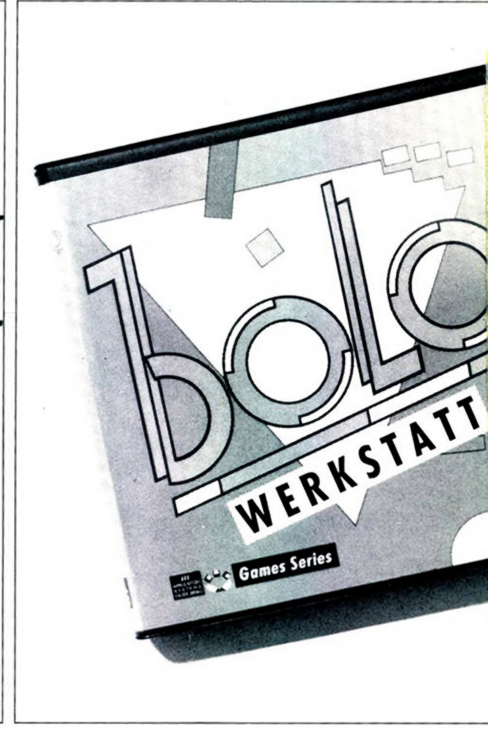

ATARI
ST
COMPUTER
Die Fachzeitschrift für den ATARI-ST Anwender.
August
September
88
8/9
DM 7,–
Ös. 56,- Sfr. 7,-
SAMPLE MIR'S
NOCH EINMAL SAM
-Soundsampler im Selbstbau
Laser C
-Megamax jüngster Bruder
DFÜ im Griff
-Telecommunikation mit
Interlink ST
Am laufenden Band
-Ein Streamer für den ST







Wie Daisy ihre Post macht.

Daisy macht es natürlich wie alle anderen: sie klebt eine Marke auf den Umschlag und der Brief geht ab in den Kasten. Das kann ihr auch kein Computer abnehmen, geschweige denn eine Software.

Dennoch: Daisy hat mit *Daily Mail* eine Software in der Hand, mit der sie spielend einen Brief mit unterschiedlichen Adressen versehen kann. Die Adressen befinden sich in einer Datenbank.
Unter anderem.

Briefköpfe können mit automatischer Absenderangabe angegeben werden. Briefe schreibt sie genauso unkompliziert wie mit einer normalen Textverarbeitung. Serienbrief und Sammelausdruck sind möglich, darüber hinaus verfügt *Daily Mail* über eine Schnittstelle zur Datenbank ADIMENS.
Zum Beispiel.

Daily Mail ist, der Autor dieser Zeilen bestätigt es hiermit, herzerfrischend einfach und logisch aufgebaut, ein Programm ohne großes Hin und Her, ideal für den täglichen Gebrauch. Mehr darüber verrät übrigens unser Software-Info „Daily Mail", das wir auf Anfrage gerne zusenden.
Aber bitte.

Lesen und lesen lassen.

Wie das so ist: Da sitzt man da und grübelt über den richtigen Software-Support. Man hat sich schon oft mit den Anwendern unterhalten, telefoniert, hunderte, ja tausende von Briefen geschrieben.

In punkto Signum! hat sich dabei in zwei Jahren eine unglaubliche Fülle von Informationen angesammelt, die den Umfang und den Inhalt jeden Handbuches sprengen würde.

Darum haben wir uns entschlossen, das definitive Buch zum Thema herauszubringen, denn wer könnte dafür kompetenter sein, als Volker Ritzhaupt, der Signum! von Anfang an betreut. Kurzum: *„Das Signum!-Buch"* ist da!

Im Format DIN A 5 auf 432 Seiten zeigt das Buch dem Anfänger alles vom ersten Schritt an, dazu praktische Beispiele und Bilder. Der Fortgeschrittene findet die kompletten Tips und Tricks und einen umfangreichen Nachschlageteil. *„Das Signum!-Buch"* ist stabil gebunden, fadengeheftet und kostet 59,– DM.

Natürlich gibt es noch ein zweites Buch, wir ahnten es schon, das enthält alle Signum!-Zeichensätze, die bisher verfügbar sind. Sie sind übersichtlich dargestellt und werden in ihrer Anwendung in der Gestaltung gezeigt. Dieses Buch ist auch von Volker Ritzhaupt und hat auch einen Namen: *„351 Signum!-Zeichensätze. Ein Buch zur Gestaltung"*. Es ist im Großformat, Paperback, hat 232 Seiten und kostet 29,– DM.

Die Bücher sind im Computerfach- und im Buchhandel erhältlich oder direkt bei uns.

Basteln da welche?

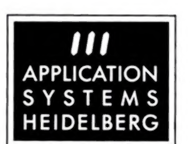
Wir haben für diejenigen eine gute Nachricht, denen *Bolo* den Schweiß auf die Stirne getrieben hat. Und wir haben eine schlechte Nachricht für die Mausartisten, die *Bolo* mit links bewältigten (angeblich, sagt unser Hans-Holger). Zum einen und zum anderen handelt es sich natürlich um die selbe Nachricht:

Jetzt gibt es die *Bolo Werkstatt*, ein flottes Utility, mit dem die ersteren endlich über das verflixte dreizehnte Level hinauszukommen in der Lage sein werden. Den anderen ermöglicht es, sich selbst so viele Hürden aufzustellen, daß sie aus dem Mausstolpern nicht mehr herauskommen und sich der gewünschte Lustgewinn beim Spielen wieder einstellt.

Alle bedauernswerten Personen, die noch gar kein *Bolo* haben, haben die Möglichkeit, sich mit der *Bolo Werkstatt* eins zu basteln (zum Spielen benötigt man jedoch weiterhin ein Original-*Bolo*).

Die *Bolo Werkstatt* gibt es überall da, wo es auch *Bolo* gibt und kostet 69,– DM.

Die Kunst der ST-Software



Englerstraße 3
D-6900 Heidelberg
Telefon (0 62 21) 30 00 02
Telefax (0 62 21) 30 03 89

Vertrieb in der Schweiz:
Computer Trend AG
Langstrasse 31
CH-8021 Zürich

EDITORIAL

Sehr verehrte Damen und Herren, liebe Leser,

in diesen Zeilen soll der Tester, genauer gesagt, das Objekt seiner Beschäftigung, der breiten Öffentlichkeit unter den Namen 'Test', 'Testbericht' usw. geläufig, zu seinem Recht kommen. Wie jedem von Ihnen bekannt, stellt ein Testbericht hohe ethische Anforderungen an seinen Verfasser, schließlich ist doch absolute Objektivität, Emotionslosigkeit vor, jedoch nicht unbedingt nach dem Test, zusammen mit einem hohen Maß an Gründlichkeit und Sachverstand die Basis unser aller Arbeit. Habe ich das nicht schön gesagt? Unser Berufsstand hat es eben sehr schwer. Ein anderer Begriff: Aktualität.

Es ist immer schön, der erste zu sein. Es verschafft den gewissen Hauch der Exklusivität, den jede Zeitung sich wünscht. Leider ergeben sich aus der wünschenswerten Aktualität auf einem Markt, der mit extrem langen Vorankündigungszeiten für extrem umfangreiche Produkte zu leben pflegt, ein paar unangenehme Konsequenzen: Wer der erste sein will, kann es sich nicht mehr leisten, auf ein Endprodukt zu warten, es sei denn, er kann sich, durch gute Verbindungen zum Hersteller, sicher sein, daß das Produkt etwas früher auf seinem Schreibtisch als auf denen der Konkurrenz ankommen wird. Die Folge: In Zeitungen häufen sich Testberichte, die entweder auf Grund von Vorab-Versionen oder sogar nur auf der Basis von kurzen Messevorführungen entstanden sind.

Es ist sicherlich auch richtig, Vorab-Berichte zu schreiben, um auf Zukünftiges aufmerksam zu machen. Jedoch betrachten wir es nicht als guten journalistischen Stil, solche Berichte als 'ersten großen Test' aufzumachen. Wir haben uns zu Beginn unserer Redaktionsarbeit entschlossen, möglichst wenig Beta-Test- oder Vorab-Versionen von Programmen zu testen, und möchten dies auch in Zukunft so handhaben; Ihre Meinung zu dieser Frage, verehrte Leser, ist uns jedoch am wichtigsten. Wir sind über jede Zuschrift zu diesem Thema sehr erfreut, also schreiben Sie uns.

Mit den herzlichsten guten Wünschen für die Sommerferien, Ihr

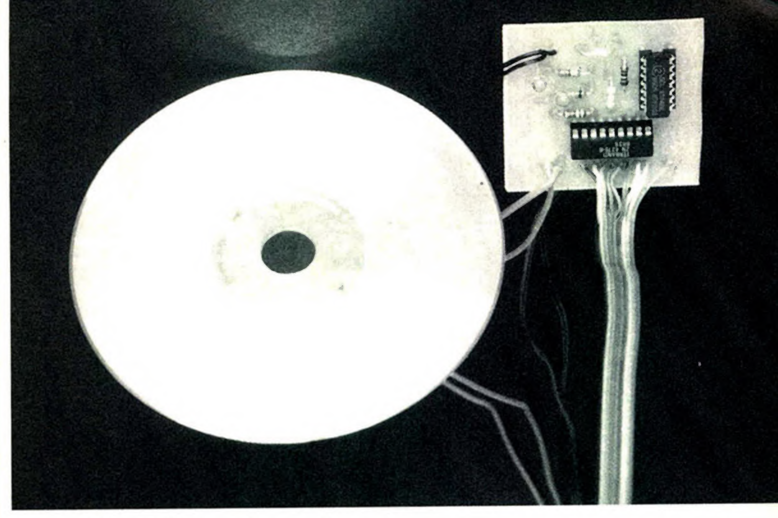
Christian Schormann

I N H A L T

SOFTWARE

HARDWARE

ANWENDUNGEN



Der Bauchredner - Sound-Sampling im Selbstbau

Eine der interessantesten und vielseitigsten Einsatzmöglichkeiten von Computersystemen ist die Erfassung und Verarbeitung von analogen Meßdaten. Zu diesen zählt man digitale Filter, Speicheroszilloskope sowie die Erfassung von Tönen aller Art (Sound-Sampling genannt). Die gestiegenen Anforderungen bei der Meßgenauigkeit vieler physikalischer Vorgänge sind ohne die Meßerfassung durch einen Rechner nicht zu befriedigen. Aber auch die moderne Unterhaltungselektronik profitiert von solcher Technik. Die schon sehr verbreiteten CD-Player verdanken ihre hohe Qualitätsstufe der Digitaltechnik. Wir bieten Ihnen deswegen einen Sound-Sampler zum Eigenbau.

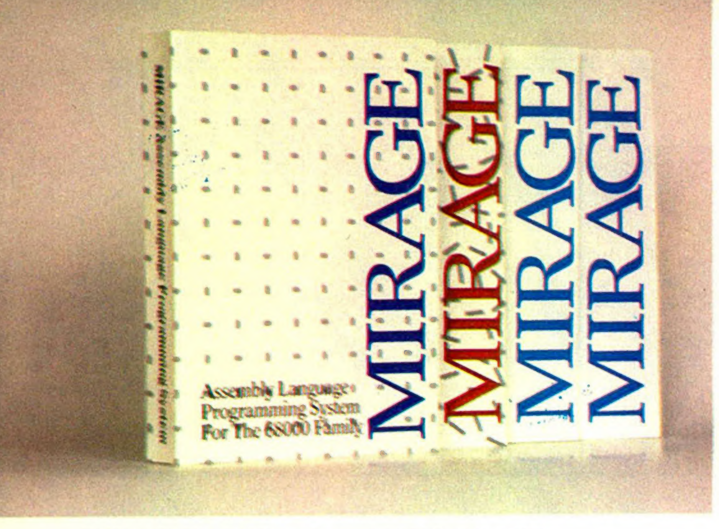
Seite **22**



Datenschlucker - Streamer im Test

Festplatten sind momentan eine echte Modeerscheinung; viele haben schon eine, fast jeder redet davon. Kein Wunder, daß auch Bedarf an Backup-Medien für die ach so empfindlichen Platten entsteht. Bisher behalf man sich, indem man den Platteninhalt regelmäßig auf Disketten kopierte. Programme für diesen Zweck gibt es einige; in der ST-Computer konnten Sie auch schon manchen Test lesen. Nun aber werden wir ST-Hacker endlich in den Hochadel der Computeranwender aufgenommen: Die Streamer kommen. In diesem Falle zwei Exemplare von der Firma Rhothron, die wir für Sie getestet haben.

Seite **16**



MIRAGE für den ATARI ST - Ein Multiuser-/Multitasking-Betriebssystem

Das in England verbreitete Multiuser-/Multitasking-Betriebssystem MIRAGE ist nun auch für den ATARI ST erhältlich. Ursprünglich für Workstations entworfen, soll es nun den Luxus einer solchen auch für den einfachen Feld-, Wald- und Wiesen-Hacker bringen - zu einem Bruchteil der Kosten. Dieser Test zeigt, ob dies gelungen ist.



SuperScore - Ein Notenlayoutprogramm

Auf der Frankfurter Musikmesse gab es an einem kleinen, in einer Ecke versteckten Stand ein interessantes Produkt zu sehen, das endlich auch Erleichterung für die handschriftgeschädigten Komponisten und Arrangeure bringen soll. Der kleine Unterschied zu den meisten anderen für den ST verfügbaren Programmen dieser Gattung ist der, der zwischen einer Textverarbeitung wie 1st_Word und einem Desktop Publishing-Programm besteht: Ersteres erlaubt es, einfach und schnell große Mengen von Text zu erfassen und in brauchbar präsentable Form zu bringen. Ein DTP-Programm dagegen soll dem Layout, der grafisch möglichst vollkommenen Fassung der meist anderswo erfaßten Texte dienen. SuperScore ist ein Programm, das es ermöglicht, auch musikalisch komplexe Noten zu gestalten und zu drucken. Für die Eingabe von Noten wurde ein 32-spuriger Sequencer gleich in das Programm integriert. Man darf gespannt sein, ob das Gespann sich bewährt.

Seite **131**

GRUNDLAGEN

PROGRAMMIERPRAXIS

AKTUELLES

RUBRIKEN

NEWS

Koprozessor SFP004 Preissensation von ATARI

Unter der Bezeichnung SFP004 ist ab sofort eine Fließkomma Koprozessor Karte für alle Mega ST Rechner zu haben. Verwendung findet der Motorola 68881 "Floating-Point-Coprocessor" Chip, der mit einer Taktfrequenz von 16 MHz betrieben wird. Die Karte wird direkt auf den internen Erweiterungsbus des Mega STs gesteckt. Aus Garantiegründen muß der Einbau durch den Fachhändler erfolgen. Zu beachten ist, daß allein durch den Einbau der Koprozessor Karte noch keine Geschwindigkeitssteigerung zu erwarten ist. Erst durch spezielle Software, die die Register des Koprozessors ansprechen, werden mathematische Routinen extrem beschleunigt. Mittlerweile gibt es schon für einige Programmiersprachen Compiler, die den 68881 unterstützen. Erstaunlich ist der Verkaufspreis von nur DM 398,-.

ATARI Computer GmbH
Postfach 1213
6096 Raunheim

Preiserhöhung in Sicht

Der anhaltende Mangel bei den Computer-Chips, vor allem bei den Speicherbausteinen den sogenannten DRAMs läßt die Preise für die begehrten Speicher weiter in die Höhe steigen. So sind die Preise für DRAMs im letzten Jahr um das 6 bis 10 fache gestiegen. Eine Besserung der Situation ist auch nicht vor dem zweiten Quartal 1989 in Sicht, so daß ATARI-Geschäftsführer Alwin Stumpf eine Preiserhöhung der ATARI Rechner ab dem 1. Oktober 1988 für sehr wahrscheinlich hält. Von diesem Mangel ist natürlich nicht nur ATARI Betroffen, auch andere Computerfirmen werden ähnlich reagieren müssen, falls sie es nicht schon getan haben.

Noch ein Drucker von Brother

Mit 360 Zeichen/sec im Datendruck arbeitet dieser Drucker, der über drei verschiedene Papierzuführungen verfügt. NLQ wird mit 75 Zeichen/sec. in 2-Pass-Technik, LQ mit 50 Zeichen/sec. in 3-Pass-Technik ausgedruckt. Als sogenannter Flachbettdrucker führt er das Papier gradlinig ein, ohne es um eine Druckwalze zu wickeln. Ausgestattet ist er mit Schubtraktor und Papier-Parkposition für problemlosen Wechsel von Endlospapier auf Einzelblatt. Mit der optionalen 2-Schacht-Einzelblattzuführung SF-300 lassen sich zwei Papierstapel oder ein Papierstapel und Briefumschläge einfach verarbeiten. Weiterhin verfügt der Drucker über Farbdruck und einen 8 Kb-Speicher. Er emuliert die EPSON EX/FX-Serie und den IBM-Proprinter XL

Brother International GmbH
Tel.: 06101/8050

Zweite ATARI Messe in Düsseldorf

Einen für jeden ATARI-Interessenten wichtigen Termin sollten Sie sich schon jetzt vormerken! In der Zeit von Freitag, den 2 September bis einschließlich Sonntag, den 4.9.88 findet auf dem Düsseldorfer Messegelände in den Hallen 1 und 2 die zweite ATARI Messe statt. All diejenigen , die letztes Jahr schon dabei waren, werden sich dieses Wochenende sicherlich rot im Kalender notieren. Denn auf dieser Verkaufsmesse, die sich zu 95 Prozent nur um den ATARI ST dreht, wird jedem ATARI Besitzer mehr geboten, als auf so manchen "großen" Messen.

Professionelle Problemlösungen wie kaufmännische Anwendungen, Textverarbeitung, Desktop Publishing, CAD/CAE-Lösungen bis zur industriellen Applikation werden im Mittelpunkt der Messe stehen. Das Rahmenprogramm bilden Workshops, bei denen täglich mehrmals wechselnde Themen intensiv erklärt bzw. vorgestellt werden. Neu in diesem Jahr wird ein großes ATARI-Forum sein, bei dem interessante Themen auch von Ausstellerseite, sowie hauseigene Präsentationen in lockerer Dialogform einem breiten Publikum zugänglich gemacht werden sollen. Schließlich wird aus dem Bereich Musik noch mehr zu hören und sehen sein als im Vorjahr. Eine Reihe interessanter Präsentationen, auch aus Übersee, sind geplant. Und wer weiß, was "Big Boss" alias Jack Tramiel und seine Crew aus Amerika noch so alles zu berichten hat?

Aber auch die ATARI 8-Bit Rechner, die komplette PC-Serie vom PC1 bis PC5 und ATARIs Transputer werden zu sehen sein. Übrigens wird auch die Merlin Computer und die Redaktion der ST Computer einen eigenen Stand haben. Dort können Sie Fragen direkt an die Redaktion stellen, Ratschläge geben oder die neuste Projekte bestaunen. Also, auf geht's nach Düsseldorf.

Vokabeln und Redewendungen lernen mit dem ST

Etwas Besonderes haben sich die Herausgeber des Englischlernprogrammes EXERCISE einfallen lassen. Durch das gesamte Programm wird der Anwender von origineller, britischer Grafik begleitet. Das bedeutet aber nicht, daß das Englischlernen zu kurz kommt. Dem Lernenden stehen 3000 Vokabeln und 2400 Redewendungen zur Verfügung. Vertippt sich der Anwender bei der Eingabe der Antwort, meldet Exercise bei weniger als drei verkehrten Buchstaben einen Rechtschreibfehler und zeigt den Fehler an. . Exercise kostet 89.- DM.

Kay Laukat Verlag Psotfach 75
2304 Laboe
Tel.: 04343/8115



NEU

NEC Pinwriter P6/P7 plus.
Mehr Schriften, mehr Speicher, mehr Möglichkeiten.

80 KB — 80 KByte Pufferspeicher. Bis zu 50 Seiten ausdrucken, ohne den Computer zu blockieren.

COLOR — Farbe einfach nachrüstbar. Carbonbänder für noch besseres Schriftbild.

24 N — 24 Nadeln für hochauflösende Grafik und exzellente Schriftbilder.

Schrift — Spezialschriften und -zeichensätze auf Steckkarten, z. B. Super-Letter Quality, OCR-B, Barcode.

P — Praktische Papier-Parkposition: Das Endlospapier bleibt auch beim Einzelblattdruck im Drucker.

7 Schriften — Sieben einzeln anwählbare Profischriften: Times, Helvette, Prestige Elite, ITC Souvenir, Draft Gothik, Bold PS, Courier.

i — Ausführliches deutsches Handbuch mit Drucker-Software (incl. PINPLOT). NEC-Hotline-Service für schnelle Informationen.

Die neuen NEC Pinwriter P6/P7 plus sind das Ergebnis konsequenter Weiterentwicklung der bewährten und zuverlässigen Bestseller P6 und P7: Einfache Bedienung, viele Schriftarten und enorm große Speicher. Beide Drucker sind blitzschnell (bis zu 265 Zeichen/Sek.) und garantieren bei einer hohen Auflösung von 360 x 360 dpi feine Grafik- und Schriftdarstellungen, die den Namen „Letter Quality“ wirklich verdienen.

NEC

Weitere Informationen erhalten Sie von: NEC Deutschland GmbH, Klausenburger Straße 4, 8000 München 80, Tel: 0 89/9 30 06-0, Telefax: 0 89/93 77 76/8, Telex: 5 218 073 und 5 218 074 nec m

EASYPREP- Nachhilfestunde auf Diskette

Das Programm EASYPREP ist von zwei Lehrern entwickelt worden. Es gibt bisher noch zu wenig Lernprogramme auf dem ATARI ST, und er wird leider noch zu oft zum Spielcomputer degradiert. EASYPREP möchte, so die Autoren, nicht einfach stur Vokabeln abfragen, sondern vielmehr durch Lektionen dem Schüler Wissenswertes beibringen, dies testen und später alles auswerten. Unter dem Schlagwort Computer Aided Education ist allerdings kein Ersetzen, sondern das Begleiten des Unterrichts wie zum Beispiel eine Nachhilfestunde gemeint, zumal der Preis des Programms ungefähr zwei Nachhilfestunden, das bedeutet 59,- DM, kostet.

EASYPREP ist ein lernpsychologisch, didaktisch und methodisch konzipiertes Programm, das zum Erlenen englischer Präpositionen für den Sekundarbereich I gedacht ist. Es zeichnet sich durch seinen strukturierten Aufbau mit progressivem Lernfortschritt, Einbindung der Lerninhalte in einen situativen Kontext und durch individuelle Steuerung der Lern-, Übungs- und Testphasen aus. Der Lernteil ist in 9 Lektionen aufgegliedert und enthält ca. 1500 der gebräuchlichsten präpositionalen Wendungen; der Übungsteil besteht aus etwa 300 Übungssätzen.

Easysoft
Jutta Gohe-Reidt
Richtpfad 23
2980 Norden
Tel.: 04931/167355

Mailbox für Sternfreunde

Weitgehend unbemerkt von der Öffentlichkeit hat am 1.5.1987 ein neues Mailbox-System den Betrieb aufgenommen. ASTRO-Mail, das, wie der Name schon andeutet, sich vornehmlich an den engagierten Amateurastronomen wendet, aber auch einiges für den interessierten Laien und andere Computeranwender zu bieten hat, ist das erste und bisher einzige astronomische Mailbox-System Deutschlands. ASTRO-Mail verfügt über diverse schwarze Bretter aus dem Bereich der Astronomie und Raumfahrt, ermöglicht das Übertragen von Bildern und stellt eine Datenbank zur Verfügung. Durch die Netmail-Funktion erhält man Zugang zu anderen Netzen, und durch Batchbefehle kann eine Mailboxsitzung automatisiert werden. Das System arbeitet mit einem erweiterten GeoNet-Befehlssatz und verarbeitet XModem oder HEX-Ausgabe. Neben verschiedenen Terminal-Emulationen stehen auch fachkundige Brettbetreuer den Usern mit Rat und Tat zur Seite.
Benutzerinformationen sind unter Beifügung von 1,30 DM in Briefmarken bei der folgenden Adresse erhältlich:

Peter Bluhm
Ginsterweg 7
D- 2121 Dahlenburg

Multisync-Umrüstung für ATARI SM 124

Die Firma Eickmann Computer bietet über eine hardwaremäßige Umrüstung (100% Softwarekompatibilität) die Möglichkeit, den SM 124 auch in mittlerer und niedriger Auflösung zu betreiben. Aufbauend auf dem bereits in der ST-Computer 5/88 beschriebenen Konzept wurde die Schaltung erweitert und verbessert. Als Ergebnis können fast alle benötigten Bauteile nun auf einer Platine untergebracht werden, wodurch ein preiswertes Anbieten möglich wird. Bemerkenswert an dieser verbesserten Version mit 2-stufigem Video-Verstärker: Der Bildschirm ist bei mittlerer und geringer Auflösung nicht mehr nur postkartengroß, sondern steht in jeder Betriebsart in voller Größe zur Verfügung. Außerdem wurde die Umschaltung bedienungsfreundlich zu den anderen Bedienungselementen gelegt. Der Preis für die Umrüstung incl. Einbau beträgt DM 248.-.

Eickmann Computer
In der Römerstadt 249
6000 Frankfurt/Main 90
Tel.: 069 / 763409

Neues für STAD

Die ersten 4 STAD-SET-Disketten von Martin Scholer sind lieferbar. Auf jeder Diskette befinden sich gepackte STAD-Bilder mit zahlreichen digitalisierten bzw. gezeichneten Motiven aus veschiedenen Bereichen. Sie eignen sich hervorragend zur Weiterverarbeitung mit dem Grafikprogramm STAD. Die Disketten kosten je DM 50,- und sind ausschließlich unter folgender Adresse erhältlich.

Peter Melzer
Rütte 10
7861 Wieden

KAWAI K1 Editor/Librarian für den ATARI ST

Das voll GEM-unterstützte Programm erlaubt eine Edierung der Single- und Multi-Parameter. Neben einer intelligenten Verbindung zweier Klänge mit Hilfe eines Interpolationsalgorithmus' ist auch ein Zufallsgenerator mit definierbarer Maske zur Erzeugung neuer Sounds im Programm enthalten. Die Bibliotheksfunktionen beinhalten das komplette Verwalten aller Sounds sowie das Speichern derselben auf Diskette. Das Programm, das auf Monochrom- und Farbmonitoren läuft, kostet 280,- DM und ist bei folgender Firma zu beziehen:

MEV - Musik & Elektronik
München

Ein weiterer Editor...

für den ST, der zum Preis von 99,- DM erhältlich ist, nennt sich PKS-Edit und wird von der Firma Pahlen & Krauss Software vertrieben. PKS-Edit ist ein schneller Editor, der vor allem das Erstellen von C-und PASCAL-Programmtexten unterstützt. Er arbeitet mit einer GEM-Benutzeroberfläche und bietet vor allem umfangreiche Suchfunktionen und eine sogenannte Crossreference. Er arbeitet mit der MEGAMAX-Shell zusammen und ediert automatisch fehlerhafte Programmtexte.

Funktionstastenprogrammierung, Klammernüberprüfung und ein Druckertreiber sind auch vorhanden. Zum Erstellen von Kreuzverweisen und zur Konvertierung von Fehlermeldungen von Compilern werden Dienstprogramme mit angeboten.

Pahlen & Krauss Software
Diefenbachstr. 32
1000 Berlin 61



Aussteller aus Europa und Übersee zeigen:

- **Alles zum Thema ATARI Computer – Software, Hardware und Peripheriegeräte.**
- **Täglich Workshops und großes ATARI-Forum mit interessanten Themen und vielen Neuheiten.**

Für professionelle Anwender, Freaks, Umsteiger und Einsteiger.

Oszilloskop und Spectrum Analyser

K-Scope und K-Spect, zwei neue Produkte der Firma Kuma Computers Ltd., ermöglichen es, den ATARI ST als Oszilloskop oder Frequenzanalysator zu benutzen. Beide Produkte arbeiten mit Hilfe eines Interfaces, welches in den ROM-Port des ST gesteckt und durch eine entsprechende Software unterstützt wird. K-Scope stellt die nötige Hard- und Software zur Verfügung, um den ST in ein Oszilloskop zu verwandeln, das im Audio-Frequenz-Bereich arbeitet. Dabei präsentiert es sich mit folgenden Daten: BNC-Anschluß, 3mv bis 30V pro Teilung einstellbar über die Software, Signal-Invertierung, Sync oder Trigger, X10-Zeitbasis-Option sowie eine 1/256-Auflösung. K-Spect verwandelt den ST in einen Nieder-Frequenz-Zwei-Kanal-Frequenz-Analysator. Dabei zeichnet es sich besonders durch folgende Daten aus: BNC-Eingang, Fmax 1 bis 35Khz in 1 Khz-Schritten, Fmin 0 bis 24 Khz, Bandbreite (Hz) 781, 390, 195, 49, 24, 12. Die vertikale Skalierung kann linear, log 3dB/div oder log 10dB/div und die horizontale linear oder logarithmisch eingeteilt werden.

K-Scope und K-Spect kosten jeweils 149,95 Pfund, wobei die K-Spect-Software einzeln zum K-Scope zu einem Preis von 89.95 Pfund erhältlich ist.

Kuma Computers Ltd.
12 Horseshoe Park Pangbourne
Bershire RG8 7JW, England
Tel.: 44-7357-4335

Libraries für Softwave-Modula 2

Die Firma Schwab Software bietet ab sofort neben dem bei Ihnen erhältlichen Softwave-Modula 2-Compiler eine GEM-Library. Diese GEM-Library wurde für den Softwave-Compiler auf dem ST sowie für TURBO-PASCAL und LOGITECH Modula-2/86 auf dem PC entwickelt, so daß eine Übernahme der Programmtexte ohne großen Aufwand möglich ist. Die Bezeichner sind in allen Versionen gleich, so daß es sogar möglich ist, TURBO-PASCAL GEM-Programme in Modula-2 GEM-Programme umzuwandeln. Zu jeder Version wird ein Handbuch mitgeliefert, worin anhand von Beispielapplikationen die Befehle erklärt werden. Die Bindings kosten für Softwave Modula-2 129,- DM, für für LOGITECH Modula-2/86 198,- DM und für TURBO-Pascal 99,- DM.

Schwab Software
Mühlbachweg 6
35 Kassel

Ein Programm für Anwälte

Für jeden Anwalt und dessen Personal ist es ein Schrecken, die Forderung eines stotternden Schuldners aufgrund zeitraubender und langwieriger, komplizierter Berechnungen errechnen zu müssen. Wer weiß nicht darüber zu klagen, wie lange gut bezahlte Kräfte an derartigen Aufstellungen arbeiten müssen und für andere, wichtigere Arbeiten in dieser Zeit nicht zur Verfügung stehen? Mit diesem Programm, das von Juristen für den Einsatz in der Zwangsvollstreckung konzipiert ist, haben Sie zu einem Preis von 139,- DM ein leicht zu bedienendes Hilfsmittel zur Hand, des Ihnen einen Teil der ehemals lästigen Arbeit innerhalb kürzester Zeit erledigt. Das Programm unterstützt die Verrechnung mehrerer Zahlungen des Schuldners nach §367 BGB und ist einsetzbar für den Fall einer titulierten Hauptforderung mit mit einem Zinstermin.

Kontaktfreudig: VME-BUS und IEC/IEEE-488-Interface

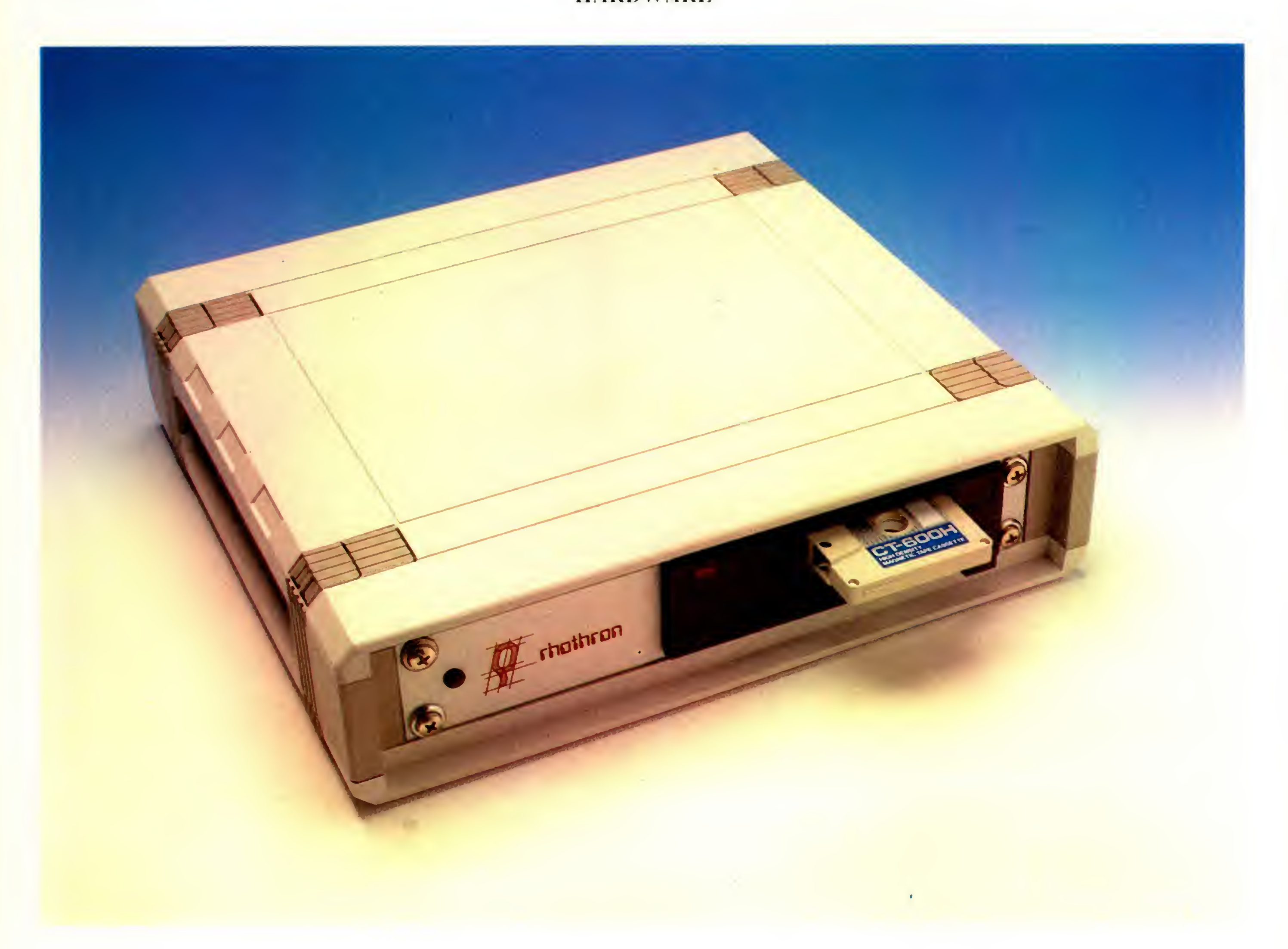
Seit der Markteinführung der ATARI MEGA-ST-Computer werden insbesondere in Forschungseinrichtungen, Laboratorien und Universitäten immer mehr Rechner dieses Typs eingesetzt. Um den Anschluß vieler Meßgeräte, die - dem weltweiten Industriestandard der Meßgerätetechnik entsprechend - mit einem IEC-Bus versehen sind, am ST zu ermöglichen, bietet die Firma GTI ein IEC/IEEE-488-Interface an. Auf dem Interface wird der GPIB-Controller-Chip NEC PD7210 verwendet. Zum IEC-Bus hin ist die Baugruppe mit Treibern ausgestattet, die die genormte maximale Übertragungsrate von 1 Mbyte/s erreichen. Das Interface belegt 256 Byte im ATARI ST und kann über vier Codierschalter im gesamten Adreßraum verschoben werden. Das Treiberprogramm beinhaltet eine vollständige Funktionsbibliothek, mit der sich alle in der Norm vorgesehenen Schnittstellenfunktionen des IEC-Busses ausführen und steuern lassen, wobei sie aus Sprachen wie C, GFA- und Omikron-Basic, PASCAL, Modula-2 etc. angesprochen werden kann.

Ein weiterer auf dem 68000er-Bus aufbauender Standard ist der VMEbus. Leider ist jedoch der Aufwand auch für kleinere VMEbus-Systeme relativ hoch, da sie mit CPU, Arbeitspeicher, Massenspeicher, Datensichtgerät und einer geeigneten Betriebssoftware ausgestattet werden müssen. Aus dieser Erkenntnis heraus hat die GTI GmbH, Berlin, eine Interface-Baugruppe entwickelt, die die Welt des VMEbusses mit der des ATARI ST verbindet. Die Baugruppe ist auf einer Doppeleuropakarte untergebracht und kann bei jedem VMEbus-System in einen freien SLOT eingesteckt werden. Das VME-ST-Interface arbeitet nach der VMEbus-Norm Rev. C1 als D16/a24-DTB-Master/Bus-Requester und beinhaltet einen über Jumper konfigurierbaren Single-Level-Bus-Arbiter. Damit kann die Baugruppe auch in Multi-Master-Systemen verwendet werden. Die Funktionen des Interfaces werden durch Firmware gesteuert, die in einem 16Kbyte EPROM untergebracht ist. Die Baugruppe besitzt einen eigenen 68000-Mikroprozessor und weitere 64Kbyte RAM. Der Treiber im ATARI ST wird resident geladen und über (erweiterte)LINE-A-Befehle gesteuert.

GTI GmbH
Unter den Eichen 108a
1000 Berlin 45

Version des BTX/VTX Managers fertiggestellt

Neben der bisherigen Unterstützung des Bildschirmtextsystems der Bundespost wird mit der Version 3.0 ein Programm geliefert, das auch das schweizer dreisprachige VTX(Videotex-System) vollständig unterstützt. Beide VTX-Dienste wie auch das französische Teletel 1, 2 und 3 sind für deutsche Teilnehmer über einen öffentlichen Gateway in Saarbrücken (0681-19302, 1200/75 Baud) zu erreichen. Die Features des Managers sind: eigene Tastaturbelegung, wie sie im Ausland üblich ist, wird unterstützt, Ausdruck über 1st_Word-Treiber ist möglich, ebenso Speicherung der Seiten ohne Informationsverlust, es ist eine unbegrenzte Anzahl von userdefinierbaren MAcros und die Version ist ISDN-vorbereitet.



DATEN-SCHLUCKER

STREAMER IM TEST

Festplatten sind momentan eine echte Modeerscheinung; viele haben schon eine, fast jeder redet davon. Kein Wunder, daß auch Bedarf an Backup-Medien für die ach so empfindlichen Platten entsteht. Bisher behalf man sich, indem man den Platteninhalt regelmäßig auf Disketten kopierte. Programme für diesen Zweck gibt es einige; in der *ST-Computer* konnten Sie auch schon manchen Test lesen. Nun aber werden wir ST-Hacker endlich in den Hochadel der Computeranwender aufgenommen: Die Streamer kommen. In diesem Falle zwei Exemplare von der Firma Rhothron.

Für PCs sind Streamer schon lange eine Selbstverständlichkeit; nicht so beim ST. Ich gehe daher davon aus, daß nur die wenigsten von Ihnen schon mit Streamern zu tun hatten. Darum wird hier nicht nur getestet, sondern auch die nötige Theorie vermittelt. Wohlan:

Was ist ein Streamer?

Kurz und scherzlos: Ein Streamer ist ein Bandlaufwerk mit sehr hoher Kapazität, mit dem man Sicherheitskopien von anderen Medien auf Bandcassetten speichert. Die meisten Streamer-Laufwerke für PCs und ähnliche Rechner lehnen sich an den *QIC-36-Standard* (*Quarter-Inch Compatibility*) an [2]: "Quarter Inch" einfach deshalb, weil die verwendeten Bänder ein viertel Zoll breit sind. QIC ist mittlerweile ein Industriestandard wie der SCSI-Bus oder das PC-AT-Steckkartenformat. Streamer mit gewöhnlichen Cassettenrecordern zu vergleichen, mag zwar manchem Puristen und Techniker frevelhaft erscheinen, ist aber recht instruktiv. Stellen Sie sich also vorerst unter einem Streamer einen Recorder mit sehr hoher Speicherkapazität vor.

Streamer & die Dire Straits

Unterschiede gibt es natürlich schon zwischen Streamern und Recordern. Bei den Cassetten wird ein anderes Bandmaterial verwendet, um den besonderen Anforderungen gerecht zu werden. Die Bandführung ist komplizierter, und insgesamt sind Standard-Streamer-Cassetten auch um einiges größer als Musik-Cassetten. Schließlich ist das Band bei Musik-Cassetten auch nur knapp 0,4 cm breit, während es bei den QIC-Bändern normaler-

weise knapp 0.6 cm (eben 1/4 Zoll) sind.

Rhothron verwendet bei den Streamern, die ich testen konnte, allerdings wiederum andere Cassetten, die tatsächlich wie normale Musik-Cassetten aussehen (siehe auch Bild 1). Auf den zweiten Blick findet man doch noch Unterschiede, nämlich den Schreibschutzschieber (langsam aussprechen und auf der Zunge zergehen lassen) und eine asymmetrisch an der oberen Kante liegende Kerbe, die diese Bänder markiert und verhindert, daß die Cassetten verkehrt herum eingelegt werden. Streamer-Cassetten kann man nicht im Laufwerk umdrehen, wie Sie das etwa mit Ihrer Dire-Straits-Aufzeichnung in Ihrem Radiorecorder machen können, wenn Sie feststellen, daß Sie eigentlich doch lieber Nicki hören. Musik-Cassetten haben diese Kerbe nicht, und deswegen kann man sie auch nicht in einen Rhothron-Streamer schieben. Modern Talking müssen Sie also schon irgendwie anders digitalisieren.

Ungleiche Brüder

Es gibt auch andere Differenzen zu den von den Rhothron-Streamern verwendeten Cassetten: Während eine Musik-Cassette nur 2*2 Spuren fassen muß (je eine Spur für den rechten und linken Kanal, und das zweimal für Vorder- und Rückseite), werden bei Streamern meist 9 Spuren aufgezeichnet. Und das Band schleicht auch nicht mit schlappen 4.75 cm/s am Lesekopf vorbei, sondern legt rasante 2 Meter pro Sekunde oder auch ein bißchen mehr vor. Das ergibt dann alles in allem eine theoretische Übertragungsrate von 720 KBit/Sekunde, immerhin fast dreimal so viel wie bei einer Floppy.

Schließlich gibt es auch noch unterschiedliche Datendichten: Hochkoerzitive Bänder, auf denen man Daten besonders eng packen kann, schlucken ohne Murren je nach Länge 50 oder 60 Megabyte, während man bei einfacheren (oder schlicht kürzeren) Bändern oder Streamer-Laufwerken auch mal mit 20 oder 30 Megabyte zufrieden sein muß.

Auf den 9 Spuren werden die Daten wieder blockweise abgelegt; die Blockgröße beträgt zumeist 512 Bytes. Eine 60-MB-Cassette faßt also etwa 120000 Blöcke. Die Aufzeichnung funktioniert - ganz ähnlich wie beim Recorder - mit einem Schreib-/Lesekopf; nur handelt es sich dabei eben um digitale (und entsprechend codierte) und nicht um analoge Daten. Daß ein Streamer ganz andere Anforderungen an den Kopf und auch an die



Bild 1: Eine Streamer- und eine Musik-Cassette

sonstige Leseelektronik stellt, sollte einleuchten.

Ich hoffe, das reicht fürs erste. Wer noch mehr über Streamer wissen will, sei auf die unten angegebene Literatur [1] und [2] verwiesen.

Zwei Streamer am ST

Die Firma Rhothron hat Streamer-Laufwerke mit 20, 40 und 60 MB Kapazität entwickelt, die man an den DMA-Bus des ST anschließen kann. Das eine Test-Laufwerk kann bis zu 20 MB auf einer Cassette aufzeichnen, das andere bringt es auf stattliche 60 Megabyte. Die 40-MB-Streamerversion, die eigenartigerweise nicht 40 MB, sondern knapp 50 MB faßt, bekamen wir nicht.

Alle diese Geräte sind noch nicht in Serie, aber laut Rhothron so gut wie fertig. Es mag sein, daß sich noch etwas an den Geräten ändert, bis sie tatsächlich verkauft werden. Möglichkeiten zur Verbesserung gibt es jedenfalls - aber dazu komme ich noch. Wir bekamen von Rhothron auch eine 60-MB-Platte zum Test, die mir aber noch ein echter Prototyp zu sein schien, ebenso wie der dazugehörige Treiber. Darum will ich mich damit nicht befassen; wenn sie wirklich fertig ist, wird sie getrennt getestet werden.

Die Hardware

Wenn Sie sich nicht sonderlich dafür interessieren, welche Klapperatismen in einem Streamer stecken, dann sollten Sie vielleicht erst wieder ab der Überschrift "Die Software" weiterlesen; für alle anderen ein paar harte Fakten:

Das Rhothron-Gehäuse des 60-MB-Streamers können Sie in Bild 2a bestaunen. Im gleichen 19"-Gehäuse war auch die 60-MB-Platte untergebracht. Es ist klein (jedenfalls kleiner als beim MegaST) und ziemlich stabil. Rhothron nennt sich ja nicht umsonst auch "Gesellschaft für medizinische Geräteentwicklung" - und in der Medizin müssen die verwendeten Geräte robust sein.

Das in Bild 2a gezeigte Gehäuse ist allerdings dermaßen solide, daß es das Laufwerk und das Netzteil darin fast hermetisch von der AuPenluft abriegelt. Nur am Lüfter findet man eine kleine Öffnung. Lüftungsschlitze gibt es nicht. Das kleine ansaugende Lüfterchenlein im Gehäuse ist zwar klar und deutlich vernehmbar (wenn ich hier so vor SH205, MegaST und Streamer sitze, komme ich mir vor wie im Seebad), doch schaufelt es wegen der ungünstigen Konvektionsverhältnisse allenfalls eine laue Brise durchs Gehäuse.

Beim Streamer ist das nicht ganz so wild, weil der nicht besonders heiß wird. Anders bei der Platte: Als ich sie nach ein, zwei Stunden Testbetrieb aufschraubte und ans Laufwerk faßte, hätte ich mir fast den rechten Mausklick-Finger verbrannt. Daß die Platte das aushält...

Das Gehäuse des 20-MB-Streamers (Bild 2b) ist dagegen etwas großzügiger - etwa MegaST-Format - und hat auch ein paar Lüftungsschlitze. Zudem ist eine Sicherung herausgeführt. Auf den Lüfter hat man hier verzichten können - eine echte Wohltat.

Welches der beiden Gehäuse für die Serienproduktion verwendet wird, blieb offen; laut Rhothron ist es wahrscheinlicher, daß für den ST-Markt das größere

Gehäuse des 20-MB-Streamers verwendet wird. Ich würde mir das auch wünschen, denn dieses Gehäuse ist zwar etwas weniger stabil, dafür aber einfach demontierbar (nur zwei Schrauben sind zu lösen) und großzügiger belüftet.

Die verwendete Hardware in den Gehäusen ist bei beiden Geräten in etwa gleich: Einen *Hostadapter* zur Umsetzung der Signale auf dem DMA-Bus des ST in ein SCSI-konformes Protokoll finden wir da. Er bietet einen durchgeschleiften und gepufferten (!) DMA-Bus, den man an der Rückseite des Gehäuses auch schön abgreifen kann. Der Wermutstropfen dabei: Rhothron verwendet nicht etwa 19-polige Stecker und Buchsen, wie das eigentlich für den 19-poligen DMA-Bus zu erwarten wäre; stattdessen werden für die etwas wacklig erscheinenden Kabel 25-polige Sub-D-Stecker eingesetzt, die ein paar Pfennige billiger sind. Von Rhothron war zu hören, daß das rein fertigungstechnische Gründe habe: Die 19-poligen DMA-Stecker gebe es bisher nur mit Lötbefestigung, während man die 25-Poler auch "anquetschen" könne. Und Löten komme bei der Produktion natürlich teurer. Wie auch immer: Ein echter Minuspunkt jedenfalls.

Warum braucht man so einen Hostadapter überhaupt? Die meisten Streamer-Laufwerke, die es so gibt, haben einen Anschluß an den *SCSI-Bus*. Das ist ein paralleler Bus für Peripheriegeräte wie etwa Platten, Streamer, CD-ROMs und Laserdrucker. ATARI verwendet am ST eine eigene Schrumpfversion dieses Busses namens ACSI. Mit den Unterschieden will ich Sie nicht quälen (siehe dazu auch [2] und [3]), es reicht in diesem Zusammenhang, wenn Sie wissen, daß vom einen Bus auf den anderen Signale umgesetzt werden müssen. Und genau das macht so ein Hostadapter. Rhothron setzt in den Streamern und in der Platte den gleichen Hostadapter ein, wenn auch in verschiedenen Entwicklungsstadien. Einzeln ist dieser Hostadapter von Rhothron nicht zu erhalten. Interessant wäre es, bietet der Hostadapter doch immerhin zwei SCSI-Anschlüsse.

Den meisten Platz im Gehäuse nimmt das eigentliche Bandlaufwerk ein. Es hat etwa die Dimensionen einer 5,25"-Floppy und würde wohl auch in einen entsprechenden Laufwerksschacht passen. Bei den Rhothron-Geräten sind TEAC-Laufwerke der MT-2ST-Serie eingebaut; neben der Mechanik und grundlegenden Elektronik bieten sie einen integrierten SCSI-Controller, in dem unter anderem ein eigener 80C85-Prozessor samt 8K-EPROM, RAM und I/O-Chips sein Unwesen treibt - im Grunde also ein eigener Rechner.

Die Hardware machte alles in allem einen zwar durchdachten, aber noch unfertigen Eindruck. Vor allem auf den Hostadapterplatinen war allerlei Ulkiges zu finden: ein abgerissener Widerstand, Huckepack-PALs, Drahtverhau, improvisierte Bestückungen auf der Platinenunterseite... so ganz serienfertig schienen die Geräte noch nicht zu sein.



Bild 2a: 20-MB-Streamer &...

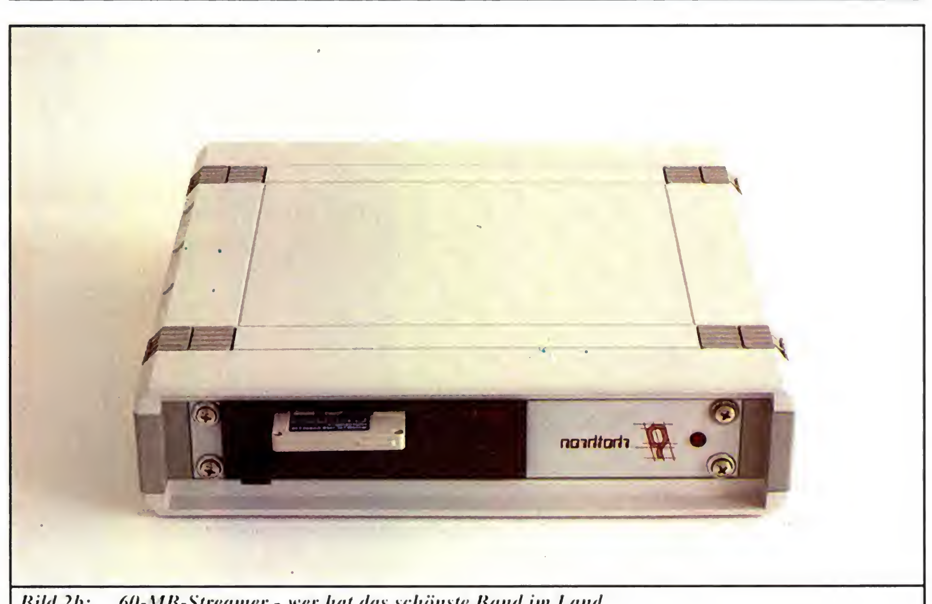


Bild 2b: ... 60-MB-Streamer - wer hat das schönste Band im Land

Die Software

Auf einer mitgelieferten Diskette waren zwei Programme versteckt, die für 40- bzw. 60-MB-Streamer zuständig sind. Für den 20-MB-Streamer gibt es anscheinend keine separate Software. Ansonsten braucht man keine Treiber oder ähnliches: also Streamer an die Platte anschließen, System hochfahren, Backup-Programm für die jeweilige Streamer-Version starten, los geht's.

Die Software präsentiert sich im hehren GEM-Gewande. Beim Start schnappt sich das Programm den gesamten verfügbaren freien Speicher, um ihn später zum Kopieren verwenden zu können. Dann werden alle Geräte am DMA-Bus durchgeprüft, was recht lange dauert (sieht so aus, als sei da eine unnütze und nervende Warteschleife versteckt!). Eine Dialogbox konfrontiert einen dann mit der bitteren Wahrheit über den DMA-Bus (Bild 3).

Kräftig gespult

Jede Cassette, die man in den Streamer legt, muß man anmelden; dabei wird das Band zum Anfang zurückgespult, um herauszufinden, was für eine Cassette da

eingelegt wurde. Gespult wird überhaupt sehr häufig, vielleicht öfter, als es sein muß. Bei der 60-MB-Cassette braucht es immerhin 2:45 Minuten, um das Band komplett vor- und zurückzuspulen. Beim Menüpunkt "Kapazität ermitteln" liest das Programm gar zur Kontrolle alle Blöcke auf dem Band ein. Dazu werden alle 9 Spuren durchlaufen - das dauert bei der 50-MB-Cassette über 10 Minuten! Warum man hier nicht einfach alle Blöcke auf *einer* Spur liest und dann die Anzahl mit 9 multipliziert, bleibt mir schleierhaft. Ich nehme mal zu Rhothrons Gunsten an, daß man so auch gleich auf defekte Blöcke auf dem Band testen wollte - aber hätte man das nicht in eine separate Option verbannen können?

Ähnliche Wartezeiten hat man vor sich, wenn man das Verzeichnis einer Cassette einlesen will. Das Verzeichnis ist nämlich nicht etwa an einer zentralen Stelle untergebracht, sondern über das ganze Band verstreut. Hier wäre durch effizientere Datenstrukturen bestimmt noch etwas zu machen.

Im Einsatz

Das Backup-Programm bietet neben allerlei Nützlichem (Band durch Vor- und Zurückspulen neu spannen, Kurzinfo über Streamer und angeschlossene Platten, Inhaltsverzeichnis der Cassette, Löschen des Bandes etc.) vor allem zwei Kopiermodi: physikalisch und logisch.

Im *physikalischem* Modus wird der Inhalt einer Platte komplett auf das Streamerband kopiert. *Logisch* heißt es bei Rhothron, wenn der Streamer eine komplette Partition oder auch eine ganze Diskette sichert. Holt man einen physikalisch geretteten Plattenzustand vom Streamer zurück, muß das Zielmedium nur formatiert, aber nicht partitioniert sein - die Partitionsinformation wurde ja mit abgespeichert.

Leider kann man physikalische und logische Kopien nicht auf einer Cassette mischen, ja, man kann nicht einmal auf eine 60-MB-Cassette mehrere physikalische 20-MB-Kopien unterbringen. Genausowenig sind Kopien von einzelnen Dateien möglich - das bietet aber schon jedes einfache Backup-Programm. So muß man also immer (mindestens) komplette Partitionen retten. Das Programm, das Rhothron dem Streamer beilegt, kopiert auch generell *alle* Sektoren einer Partition und nicht etwa nur die belegten.

Meine SH205 ist mit einigen Tricks auf mehr als 22 MB hochformatiert (wie man das macht, steht in [2] und [3]). Genauer: Es sind 22542 KB. Der 20-MB-Streamer behauptet von sich, daß er auf dem Band mehr als 23000 KB unterbringen könne: trotzdem weisen die Backup-Programme von Rhothron jeden Kopierversuch mit "Kapazität nicht ausreichend" ab. Bug or feature? Ich tippe mal auf einen Fehler, der aber schleunigst abgestellt werden sollte. Wer hätte sonst auch Bedarf für den 20-MB-Streamer?

Jetzt sind Sie aber sicher gespannt auf die Zeiten für das Hin- und Herkopieren. Sie sind in Tabelle 1 gesammelt. Rückspulzeiten sind separat angeführt. Zum Vergleich habe ich auch Kopierzeiten für das Backup-Programm Harddisk Utility (Version 2.0) von Application Systems Heidelberg angeführt - dabei habe ich aber die Diskettenwechselzeiten nicht eingerechnet.

Beim Kopieren mit dem Streamer stand jeweils etwa 1.2 MB Systemspeicher zur Verfügung. Bei kleinerem Speicher müssen mehr Durchgänge eingelegt werden: bei 400 KB freiem Speicher etwa darf man noch einmal etwa 5 bis 10 Prozent mehr Zeit veranschlagen. Gemessen wurde auf einem MegaST2.

Die Zeiten für den Restore-Vorgang (Zurückkopieren der Daten vom Streamer auf die Platte) unterscheiden sich nicht wesentlich von denen beim Sichern, nur daß die Spulerei sich etwas anders verteilt.

Der Vergleich mit dem Backup-Programm HDU 2.0 von Application Systems hinkt etwas. Wie gesagt: Beim Rhothron-Streamer werden mindestens komplette Partitionen gerettet: beim HDU und bei anderen Disketten-Backup-Programmen kann man sich aussuchen, welche Dateien gesichert werden. Komprimierungsalgorithmen fehlen in der Rhothron-Software ebenso wie eine Routine, die alle belegten Sektoren einer Partition sucht. Es sind einfach zwei völlig verschiedene Anwendungszwecke. Ein Disketten-Backup ist ideal für einzelne Partitionen und Dateien. Streamer haben dagegen den Vorteil, daß man sich nicht mit einem Haufen Disketten herumschlagen muß. Und während der Streamer kopiert, kann man die Zeit anderweitig nutzen, ohne wie beim Disketten-Backup ständig vorm Rechner auf die nächste Aufforderung zum Diskettenwechsel lauern zu müssen. Was ich damit sagen will: Eigentlich habe ich in der Tabelle Äpfel mit Birnen verglichen.

Sie sehen aber immerhin aus der Tabelle, daß der Geschwindigkeitsvorteil des Streamers schnell schrumpft, wenn viel gespult werden muß - also wenn das Band nicht ganz leer ist.

Noch ein paar Worte zur Zuverlässigkeit: Rhothron gibt an, daß bei jedem Schreibvorgang die Information auf dem Band zur Kontrolle gelesen wird, um Fehler zu erkennen. Darum habe ich im Vergleich auch HDU mit aktivierter Schreibkontrolle laufen lassen. Außerdem bemerkt das Backup-Programm auch physikalische Fehler auf dem Quellmedium, bricht seine Arbeit aber nicht ab, sondern meldet hernach die Anzahl der Fehler. Fehler im

Bild 3: Die Streamer-Software meldet sich zum Rapport - hier ohne angeschlossenen Streamer

Vorgang	Zeit insgesamt	davon Umspulzeit
20-MB-Platte -> 60-MB-Streamer	5:50	keine (Band war an den Anfang gespult)
6143 KB (Partition C, Band war leer) -> 60-MB-Streamer	1:45	- (Band war leer)
8191 KB (Partition D) -> 60-MB-Streamer	4:25	2:30 (auf dem Band war bereits eine Partition)
5119 KB (Partition E) —> 60-MB-Streamer	4:35	3:05 (zwei Partitionen bereits auf dem Band)
3085 KB (Partition F) —> 60-MB-Streamer	6:05	5:05 (drei Partitionen bereits auf dem Band)
6143 KB (Partition C) -> 20-MB-Streamer	1:45	- (Bandanfang)
8191 KB (Partition D) -> 20-MB-Streamer	4:45	2:25 (eine Partition bereits auf dem Band)
5119 KB (Partition E) -> 20-MB-Streamer	5:00	3:30 (zwei Partitionen auf dem Band)
3085 KB (Partition F) -> 20-MB-Streamer		4:50, danach Abbruch mit "Kapazität nicht ausreichend"
Inhaltsverzeichnis einlesen	4:48	4:48 (vier Partitionen auf dem Band)
Umspulen bei 50-MB-Cassette		2:30
Umspulen bei 60-MB-Cassette		2:45
Cassette löschen		2:30 (50 MB) bzw. 2:45 (60 MB)
Kapazität bestimmen (50 MB)		10:20
Kapazität bestimmen (60 MB)		>10 min, irgendwann abgebrochen
3.3 MB per HDU 2.0 auf Diskette (mit Verify, TURBODOS + FASTLOAD, 9-Sektor-Diskette)	6:45 (*)	Diskettenwechsel nicht gemessen (etwa 10 s pro Diskette)
das gleiche ohne TURBODOS	7:15	Diskettenwechsel nicht gemessen (etwa 10 s pro Diskette)

(*) pro 720-KB-Diskette brauchte HDU etwa 1:30; damit können Sie sich leicht ausrechnen, wie lange man braucht, um damit soundsoviel MB abzuspeichern. Alle Zeitangaben übrigens in Minuten:Sekunden.

Tabelle 1: Zeit ist Geld - wie schnell sind die Rhothron-Streamer?

Streamer-Laufwerk (Cassettenwechsel, Schreibschutz etc.) werden sauber erkannt und in deutschem Klartext gemeldet.

Ein bißchen Probleme hatte der Streamer aber offensichtlich, wenn die Wege am DMA-Bus zu lang wurden. So hatte ich etwa beim Test folgende Konfiguration: DMA-Kabel vom MegaST zur SH205, "langes" Rhothron-Kabel von der SH205 zur Rhothron-Platte, kurzes Kabel von der Rhothron-Platte zum Streamer. Jeder Versuch, bei dieser Konfiguration von der SH205 auf den Streamer zu kopieren, scheiterte mit "Fehler 32 - DMA-Fehler Harddisk". Im Handbuch steht dazu nur die lakonische Bemerkung, ich solle doch bitte den Hersteller benachrichtigen.

Nun ist es aber bekannt, daß der DMA-Bus des ST nicht besonders stabil ist (weil ungepuffert); dazu kommt, daß nur die wenigsten zwei Platten gleichzeitig angeschlossen haben. Und wenn man sie doch hat, kann man das Problem - wie auch in meinem Fall - durch Umstecken umgehen. Deswegen bleibt der Fehler zwar unschön, kann aber nicht eindeutig dem Streamer angelastet werden. Unangenehm an der Geschichte ist nur, daß man beim Umstecken ständig in Konfusion gerät, weil die Rhothron-Geräte stur auf ihren 25-poligen Anschlüssen beharren.

Das Handbuch

Leider merkt man dem Handbuch an, daß es von Technikern geschrieben wurde, die nicht besonders viel Spaß am Formulieren haben. Zwar findet man auf knapp 30 Seiten meistens, was man wissen will; auch ist jeder Befehl im Menü einzeln erläutert (samt Bildschirmkopie). Doch Freude will beim Durchblättern nicht aufkommen; das ist halt wieder eine von diesen trockenen, typisch deutschen Anleitungen: "Der Netzanschluß des Streamers erfolgt über das mitgelieferte Netzkabel" (ach nee!). Man entblödet sich auch nicht der typisch anglizistischen Bindestrich-Sparwut: "Defekt des Streamer Laufwerks." In einem Leserbrief in irgendeiner Ausgabe der Computerzeitung c't hieß es dazu treffend: "Das stört den Lese Fluß." Im Deutschen schreibt man halt "Streamer-Laufwerk" oder auch "Streamerlaufwerk"; alles andere ist schlicht falsch.

Aber auch beim Handbuch hat Rhothron ja noch Gelegenheit, es zur endgültigen Serienreife zu bringen. Merke: Ein gutes (also nicht ATARI-kompatibles) Handbuch nimmt den Kunden von vorneherein für den Hersteller ein. Aber immerhin ist es ja deutsch, und es steht alles Wichtige drin. Vielleicht bin ich da auch ein bißchen pingelig oder altmodisch oder beides...

Alles in allem

Die Streamer tun ihren Dienst, ohne daß man sich über größere Betriebsfehler ärgern müßte. Bis auf die Schwierigkeiten mit dem DMA-Bus, für die Rhothron aber wahrscheinlich nichts kann, hatte ich keine Probleme damit. Die eigentlichen Kopierzeiten sind auch recht ansehnlich; 5:50 für 20 MB sind ja wahrlich nicht schlecht. Nur muß man fairerweise auch die vielen Umspulzeiten mit ins Kalkül ziehen - und dann sieht die Rechnung schon nicht mehr so gut aus. Einiges von diesem Verwaltungsaufwand könnte man sich meiner Meinung nach sparen. Vor allem, wenn man logische Kopien anfertigt (also Partitionen kopiert), übersteigen die Umspulzeiten leicht die eigentliche Kopierzeit.

Dazu kommt, daß die Software leider auch nicht so intelligent ist, nur die belegten Blöcke zu kopieren. Dateiweises

Kopieren bleibt hier ohnehin nur Wunschtraum. Vielleicht ist das bei einem Streamer ja auch nicht so besonders sinnvoll, wegen ein paar Dateien von 100 oder 200 KB die Umspulzeiten in Kauf zu nehmen - vielleicht aber doch. Denn fehlt eine solche Option im Backup-Programm des Streamers, muß man doch wieder auf Disketten umsteigen. Bei solchen kleineren Kopieranwendungen (Kopieren von teilweise gefüllten Partitionen oder von Dateien) sind Backup-Programme für den Diskettenbetrieb nämlich letztlich flexibler und zuweilen auch schneller - zumindest im vorliegenden Falle. In Frage kommen die Rhothron-Streamer also wohl nur für große Datenmengen.

Otto ST-Normalanwender wird sich wahrscheinlich die Preise für die Streamer ansehen: 2100 DM für 20 MB, 2300 DM für 40 MB, 2500 DM für die große 60-MB-Version. Ob er sich dann kopfschüttelnd und mit leisem Bedauern abwendet, sein Backup-Programm startet und seinen Diskwechsel-Bizeps trainiert, oder ob er wohl begeistert loszieht und sich gleich einen Zwölferpack Streamer (oder vielleicht auch nur einen davon) besorgt? In der Regel ersteres - vermute ich.

Wer aber beruflich an ST und Platte arbeitet und oft große Backups machen muß, der spart eine Menge Zeit - und die ist ja bekanntlich auch Geld. Letztlich ist das dann eine kaufmännische Entscheidung. Hätte ich mehrere Arbeitsplätze mit mehreren STs und Festplatten, wäre für mich die Sache klar: Ein Streamer muß her. Eine andere Anwendung: Größere Datenmengen, die laufend erfaßt werden, müssen regelmäßig platzsparend archiviert werden (Personaldaten, Börsenkurse...). Immerhin passen auf eine 60-MB-Cassette soviele Daten wie auf drei SH205-Platten. Ist das nichts? Übrigens: Eine 60-MB-Cassette kostet unter 70 DM.

Die Rhothron-Streamer sind allerdings noch nicht 100%ig ausgereift und haben eine Nachbehandlung nötig, bevor sie in Serie gehen - vor allem bei der Software. Welche Verbesserungen eventuell eines schönen Tages in die Entwicklung einfließen, werden Sie rechtzeitig in den News erfahren.

Claus Brod

Bezugsadresse:

Rhothron
Gesellschaft für medizinische Geräte- und Systementwicklung mbH
Entenmühlstraße 57
D-6650 Homburg/Saar
06841/64067

Literatur:

[1] *Western Digital: Storage Management Products Handbook 1986*

[2] *Claus Brod, Anton Stepper: SCHEIBENKLEISTER - Massenspeicher am ST, Eschborn 1988*

[3] *Hans-Peter Labude: Die Festplatte, Serie in der ST-Computer (1/88 bis 6/88)*

ENDE



Daten lassen sich mit Hilfe von Brainstorm gemäß ihrer hierarchischen Struktur ordnen. Leicht zu bedienen und voll GEM unterstützt.
DM 149,—

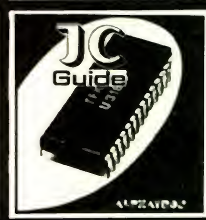


68881 Coprozessor
ERWEITERUNGSKARTE
ALPHATRON COMPUTERSYSTEME

für Atari ST **DM 698,—**
für Amiga **DM 798,—**
für MacIntosh **DM 798,—**

68881 Coprozessor Platine ab 16 MHz auf 20 MHz erweiterbar; eine Steigerung der Rechenleistung bis zum Faktor 900 ist erreichbar.

Datenbank mit über 800 IC's, die durch den Anwender erweiterbar ist.



ST Version **DM 149,—**
MS-DOS Version **DM 249,—**



Diese flexible Datenbank nach dem Karteisystem ist leicht zu bedienen und universell einsetzbar.
DM 149,—

ALPHATRON
SCHACHPROGRAMM Hamlet II
für ATARI ST

ALPHATRON COMPUTERSYSTEME
loewenichstr. 30 - d - 8520 erlangen
telefon 09131 / 2 50 18
telex 62 97 65 atron d

Hendrik Haase Computersysteme präsentiert:

Atari-Computer

1040 STF incl. SM124	1450,– DM
Atari Mega ST2 incl. SM124	2598,– DM
Atari Mega ST4 incl. SM 124	3598,– DM
Drucker Star LC 10	698,– DM
Drucker NEC P6 plus	1698,– DM
NEC Multisync II	1449,– DM
Graustufen Multisync	549,– DM
Vortex-Festplatten	a. L.

Bestellungen und Informationen bei:

Hendrik Haase Computersysteme
· Wiedfeldtstraße 77 · D-4300 Essen 1
Telefon: 02 01 - 42 25 75 · Fax: 02 01 - 41 04 21

PADERCOMP

Computerhandelsges. mbH & Co. KG
Breslauer Straße 25 · 4790 Paderborn · Telefon 0 52 51 - 77 07

Floppylaufwerke

PADERCOMP FL-1 3,5", NEC FD 1037 A-Laufwerk, anschlußfertig für ATARI ST, graues Metallgehäuse, Abm: 153 × 103 × 26 mm	**348.-**
PADERCOMP FL-2 Doppelstation übereinander, sonst wie FL-1	**598.-**
NEC FD 1037 A 3,5", 1 MB, 25,4 mm Bauhöhe, 5V	**185.-**
TEAC FD-55-FR 1 MB, 5,25", 80 Spuren, Slimline	**249.-**

Drucker

NEC P6 24 Nadeln, 216 Z/s, DIN A4	**1198.-**
NEC P7 24 Nadeln, 216 Z/s, DIN A3	**1498.-**
NEC P2200 24 Nadeln, 168 Z/s, DIN A4, P6-kompatibel	**899.-**
Epson LX-800 9 Nadeln, 180 Z/s, DIN A4	**625.-**
Epson LQ-500 24 Nadeln, 180 Z/s, DIN A4	**899.-**
Epson LQ-850 24 Nadeln, 264 Z/s, DIN A4	**1375.-**
Star LC-10 9 Nadeln, 144 Z/s, DIN A4	**575.-**
Seikosha SL 80 AI 24 Nadeln, 135 Z/s, DIN A4	**819.-**

Monitore

EIZO Flexscan 8060 S 820 × 620 Punkte, 0,28 dot pitch	**1498.-**
Mitsubishi EUM 1471 A 0,31 dot pitch, Videoeingang	**1298.-**
ATARI SM 124	**438.-**

Zubehör

FL-1-Adapter zum Anschluß von 2 FL-1	**49.00**
Multimatic Cut Sheet Feeder für P6	**598.00**
Dataphon S 21 d-2 300 Baud, FTZ-Nr.	**238.00**
Dataphon S 21/23 300 bzw. 1200/75 Baud, Btx	**319.00**
Disk Box für 50 Stück 3,5"-Disketten	**19.90**
Media Box 1 für 150 Stück 3,5"-Disketten	**34.90**
Druckerkabel ST	**34.90**
Druckerkabel P2200 an ST	**65.00**
Monitor-Ständer dreh-, schwenk- und kippbar	**25.00**
Monitor-Stecker für ATARI ST	**6.90**
Kabel EIZO/NEC an ATARI ST	**75.00**
Monitor-Umschaltbox	**47.00**
Vortex HD Plus 20 MB	**1148.00**
Btx-Term ST incl. Schnittstelle und Software	**209.00**
Preisliste	kostenlos

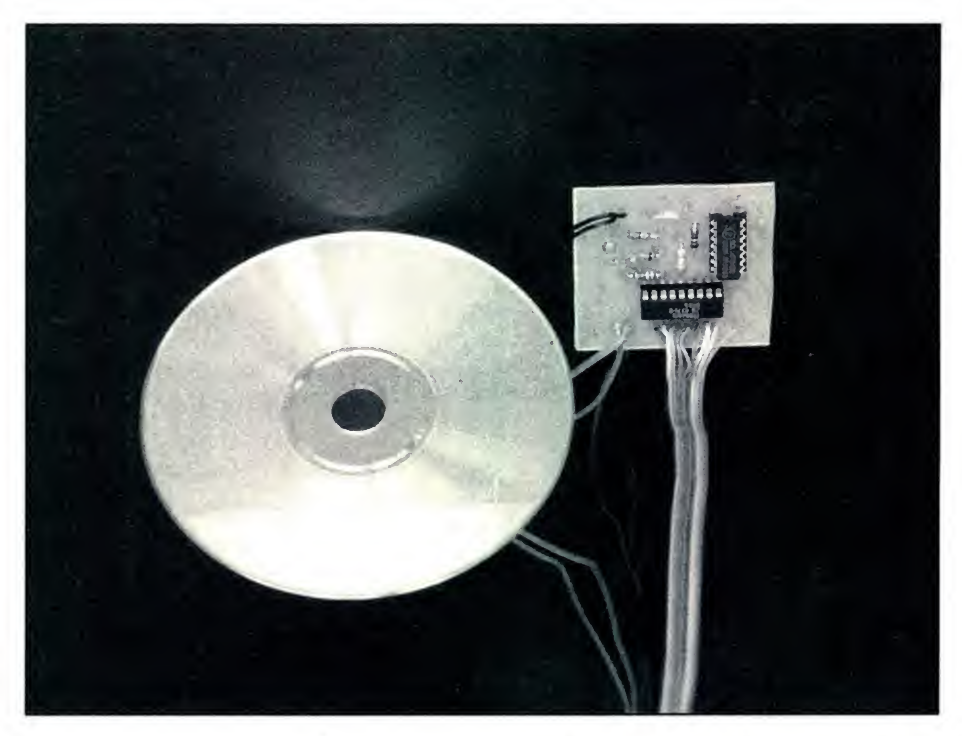
Die aktuellen Preise erfragen Sie unter der PCB-Mailbox-Tel. 0 52 52 / 34 13

Händleranfragen erwünscht

SAMPLE MIR'S NOCH EINMAL SAM

Soundsampler im Selbstbau

Eine der interessantesten und vielseitigsten Einsatzmöglichkeiten von Computersystemen ist die Erfassung und Verarbeitung von analogen Meßdaten. Zu diesen zählt man digitale Filter, Speicheroszilloskope sowie die Erfassung von Tönen aller Art (Sound-Sampling genannt). Die gestiegenen Anforderungen bei der Meßgenauigkeit vieler physikalischer Vorgänge sind ohne die Meßerfassung durch einen Rechner nicht zu befriedigen. Aber auch die moderne Unterhaltungselektronik profitiert von solcher Technik. Die schon sehr verbreiteten CD-Player verdanken ihre hohe Qualitätsstufe der Digitaltechnik.



Allen diesen Anwendungsmöglichkeiten liegt das gleiche Prinzip zugrunde: die Umsetzung von elektrischen Spannungen als "Rohmaterial" in digitale Daten mittels mathematischer Berechnungen. Aber kein Rechner kann Analogsignale verarbeiten. Aus diesem Grund wird ein analoges Signal (stetiger Spannungsverlauf) an den Eingang eines Umsetzers, des sogenannten A/D-Wandlers geführt, der dann an seinem Ausgang dem Rechner ein digitales Signal in Form einer Zahlenkombination zur Verfügung stellt.

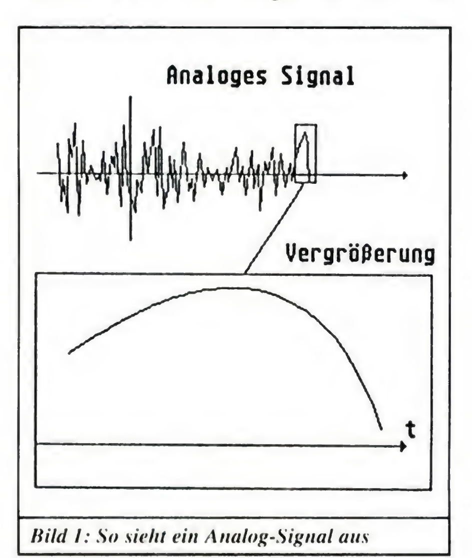


Bild 1: So sieht ein Analog-Signal aus

Das hier beschriebene "Sound Sampling" ermöglicht es, Sprache oder sogar ganze Musikstücke mit einem ATARAI-ST zu digitalisieren, zu bearbeiten und über den Monitorlautsprecher wieder auszugeben. Selbstverständlich darf man von diesem Sound Sampler nicht HiFi-Qualität erwarten, aber um z.B. einen Programm-Vorspann noch mit Musik zu versehen, reicht sie aus.

Für den interessierten Anfänger, der noch nicht weiß, was ein Sound Sampler ist, oder was der Begriff 'digitalisieren' bedeutet, sollen diese jetzt erklärt werden. Was ein Sound-Sampler tut, ist schnell gesagt. Er wandelt die Signale, die ein handelsüblicher Kassettenrekorder, ein Radio oder ein ähnliches Gerät über den Kopfhörerausgang, den Lautsprecheranschluß oder ein Mikrofon ausgeben (siehe Bild 1), in für den Computer verständliche Signale um. Da aber ein Computer nichts anderes versteht als Zahlen, muß ein solches Signal in Zahlen umgewandelt werden. Diese Aufgabe übernimmt ein sogenannter Analog-Digital-Wandler. Im Prinzip besteht ein Sound-Sampler aus nichts anderem als einem A/D-Umsetzer. Analog nennt man das Signal, das z.B. ein Kassettenrekorder ausgibt; digital das Signal, das vom Computer verstanden werden kann - Zahlen. Digitalisiert man also etwas, so wird ein Signal in ein für den Computer verständliches verwandelt.

Quantisierung & Abtastung

Um ein Analogsignal (z. B. einen stetigen Spannungsverlauf) in ein digitales umzuwandeln, muß es in bestimmten Abstän-

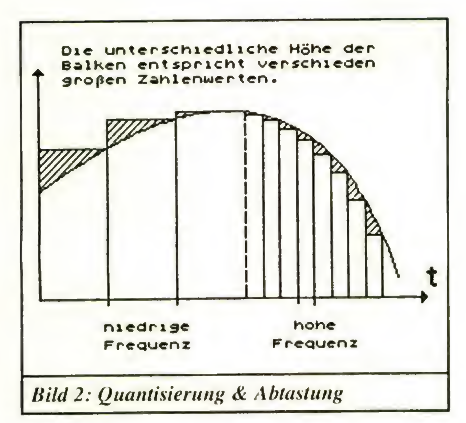


Bild 2: Quantisierung & Abtastung

den abgetastet und dem Spannungswert zur Zeit der Abtastung ein Zahlenwert zugeordnet werden. Diesen Vorgang nennt man **Quantisierung**. Der zeitliche Abstand zwischen zwei Meßpunkten (Umwandlungen) wird **Abstastfrequenz** genannt. Je höher sie ist, umso kürzer sind die immer gleichen Abstände, und desto besser ist die Wiedergabequalität vom Computer. Bild 2 soll dies verdeutlichen. Der schraffierte Teil zeigt die Ungenauigkeit, die bei höherer Abtastfrequenz immer kleiner wird. Die Erfassung eines Analogsignales bzw. seine Verarbeitung in einen digitalen Wert nimmt eine gewisse Zeit in Anspruch. Ändert sich während dieser Zeit das analoge Eingangssignal, so ist mit einem Fehler bei der Umsetzung zu rechnen. Abhilfe würde ein sogenannter "Sample and Hold" am Eingang des A/D-Wandlers schaffen. Durch diesen "Sample and Hold" wird die Eingangsspannung (Probe) für die Dauer der Umwandlung gespeichert, so daß sich das Eingangssignal ändern kann, ohne daß dies einen Einfluß auf die Analog/Digital-Wandlung hat. Ein solcher "Sample and Hold" ist bei dem hier vorgestellten Projekt nicht vorgesehen, da einerseits ein solcher Wandler sehr teuer ist, anderseits für unsere Anwendung ein "normaler" 8-Bit-Wandler vollkommen ausreicht. Ein weiterer wichtiger Faktor für die Qualität einer Digitalisierung ist die **Auflösung**, mit der das analoge Signal abgetastet (zerlegt) werden kann. Bei unserem 8-Bit-Wandler kann das Eingangssignal in 2^8 = 256 Schritte (Auflösungsstufen) unterteilt werden.

Beispiel: Ein Signal mit einer Amplitude von 1 Volt_{SS} kann mit einem 8-Bit-Wandler in 256 Spannungsstufen unterteilt werden, d.h. man hat eine Auflösung von 1/256 gleich 3,91 mV (Millivolt).

Da nun die meisten Computer von der Hardware her nicht in der Lage sind den Digitalisiervorgang durchzuführen, genügt es nicht, einfach die entsprechende Software einzuladen, sondern man muß sich auch einer entsprechenden Schnittstelle zur Außenwelt bedienen, die diese Arbeit erledigt. Ein solches Interface, einen "Sound Sampler" stellt die hier gezeigte Schaltung dar.

Die Hardware

Das Herz des Samplers ist ein etwa 35,- DM teurer A/D-Wandler mit der Bezeichnung ZN427, der problemlos erhältlich ist und auch ohne eine allzu aufwendige Zusatzschaltung auskommt. Der ZN427 ist ein bekannter, leistungsfähiger und schneller 8-Bit-Wandler, der zudem relativ preisgünstig ist. Die maximale Eingangsspannung beträgt 5 Volt. Hier nun die Beschreibung der Anschluß-Pins des ZN427:

Pin 1 **Busy:** wird während des Wandlungsvorganges auf Low gesetzt, damit der Rechner die Daten als ungültig erkennen kann.

Pin 2 **Output Enable:** bei Low gehen die Datenausgänge in den hochohmigen Zustand.

Pin 3 **Clock-Eingang:** regelt den internen Wandlungsvorgang, wobei pro Wert 9 Taktimpulse benötigt werden (8+1).

Pin 4 **WR:** Die Wandlung wird gestartet, wenn der Rechner diesen Eingang auf Low zieht.

Pin 5 **R.ext.:** hier liegt die negative Versorgungsspannung an.

Pin 6 **U in:** Eingang für die analogen Signale.

Pin 7 **REFin:** bestimmt als Referenzspannungseingang den Wandlungsbereich.

Pin 8 **Refout:** Ausgang für Referenzspannung (wird mit REFin verbunden).

Pin 9 **GND:** Masseeingang.

Pin 10 **Vcc:** Eingang für Spannungsversorgung von +5 Volt, die aus dem Joystickport abgenommen werden.

Pin 11-18 **Ausgänge:** Von denen aus die digitalen Daten an den Computer geschickt werden.

Die übrigen Bauteile der Schaltung dienen als Oszillatorstufe und der Erzeugung der negativen Versorgungsspannung. Da deren Zusammensetzung relativ einfach gehalten ist, erübrigt sich eine ausführliche Beschreibung.

Der Aufbau

Die gesamte Schaltung des A/D-Wandlers findet auf einer Platine von etwa 10x5 cm Platz, die nur einseitig beschichtet sein muß. Wer sich das Aufzeichnen des Layouts ersparen möchte, kann auch eine Lochrasterplatine verwenden. Die gedruckte Schaltung sieht jedoch besser aus, ist übersichtlicher und dadurch leichter zu bestücken.

Beim Einlöten der Bauteile dürften eigentlich keine Probleme entstehen, lediglich die beiden ICs sollten gesockelt werden.

Ist die Platine bestückt, verbindet man die Datenausgänge und die Strobe-Leitung sowie die Masse durch ein Flachbandkabel mit den entsprechenden Pins eines 25-poligen D-Sub-Steckers, der am Druckerport des ATARI-ST angeschlossen wird. Da dieser Port keine Versorgungsspan-

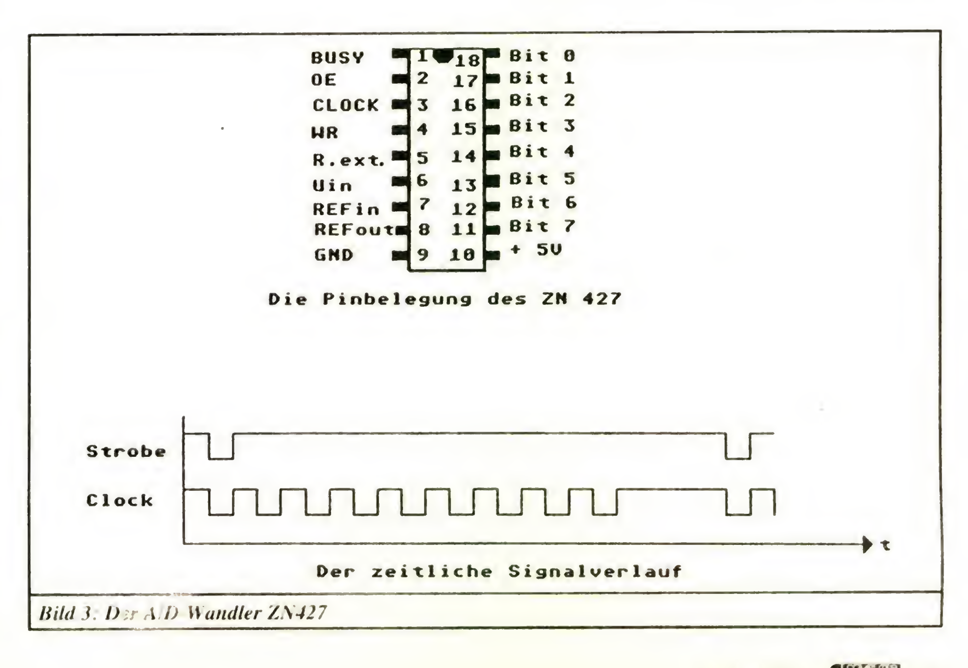


Bild 3: Der A/D Wandler ZN427

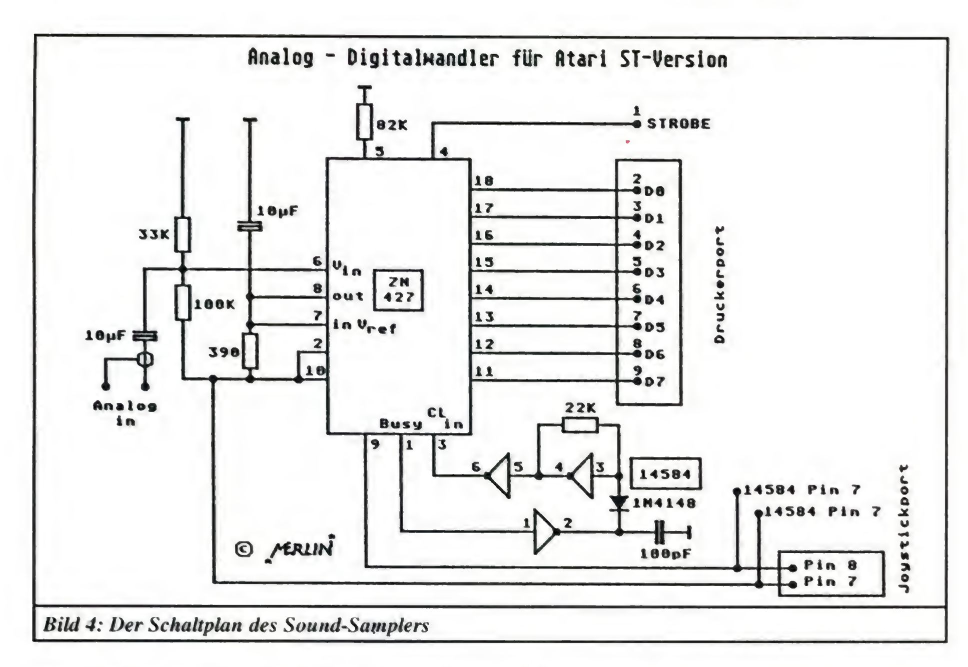


Bild 4: Der Schaltplan des Sound-Samplers

nung liefert, greift man diese am Joystickport Pin Nr.7 ab. Hierbei ergibt sich allerdings das Problem der Steckverbindung, da die Seriensteckcr nicht in den Port passen, und bei Joystickkabeln der Pin 7 nicht belegt ist. Ich habe daher von einem Serienstecker die Blechummantelung abgenommen und die Kunststoffteile wieder zusammengeklebt. An Pin 7 liegen +5V an. Ein Stück Flachbandkabel, das blind an einige Pins angelötet wird, dient der Zugentlastung.

Zum Schluß muß noch der Stecker für den Analogeingang angelötet werden. Da es hier verschiedene Typen und Größen gibt, muß man sich dabei nach dem individuellen Ausgabegerät (Kassettenrekorder, Stereoanlage) richten. Jedenfalls muß der Stecker in einen der Ausgänge passen. Ist nun das Gerät fertig zusammengebaut, so läßt es sich folgendermaßen testen:

Man schließt Versorgungsspannung und Analogeingang an und mißt mit einem Multimeter die Spannungswerte am Sockel des ZN427, dabei dürfen noch keine ICs eingesteckt sein.

Pin 2: 5V
Pin 5: -3V
Pin 6: 2,5V
Pin 9: 0V
Pin 10: 5V

An Pin 3 sollte das Oszilloskop eine Rechteckschwingung zeigen. Sind die Werte in Ordnung, steckt man die beiden Chips in die Sockel (auf die Kerbung achten) und schließt noch den D-Sub-Stecker am Druckerport an (im ausgeschalteten Zustand des Rechners), lädt die Software und ab geht die Post.

Da der Monitorlautsprecher des ATARI-ST hohe Töne nicht übertragen kann, erweist sich eine Abtastfrequenz von 20kHz als die beste Lösung, zumal damit bei reservierten 680 KByte stolze 35 Sekunden gesampelt werden können.

Die Software

Der Aufwand der bis hierhin getrieben wurde, ermöglicht es noch nicht, eine digitalisierte Stimme aus dem Lautsprecher des SM 124 hören zu können. Dafür benötigt man noch etwas Software, die die gesamten Informationen (Daten) richtig aufbereitet und wiedergibt. Das hier aufgelistete Assembler-Programm ist die minimale Version eines Sound-Samplers, um Geräusche, Stimmen oder auch Musik aufzunehmen bzw. wiedergeben zu können.

Dank des ausführlich dokumentierten Source Codes ist es wohl kaum nötig, die Arbeitsweise des Programms selbst zu erläutern. Auch das Auswahlmenü erklärt sich eigentlich von selbst. Dennoch soll auf dieses genauer eingegangen werden.

Mit F1 wird der Menüpunkt 'Hören' aufgerufen. Hier hat man die Möglichkeit, sich das zu sampelnde Stück schon einmal vorher anzuhören, ohne es gleich in den Speicher zu übernehmen. So lassen sich der beste Anfang und das beste Ende für das Stück leichter finden oder der Klang bei verschiedenen Frequenzen testen. Durch einen Druck auf die Space-Taste kehrt man wieder zum Auswahlmenü zurück.

Mit der F2-Taste kann man ein Stück aufnehmen. Zunächst wird das Stück lediglich gespielt, aber noch nicht in den Speicher übernommen. Durch einen Druck auf die Space-Taste beginnt der eigentliche Aufnahmevorgang. Auf diese Weise läßt sich der Anfang des Stückes sehr genau treffen. Dann wird solange aufgenommen, bis entweder der Speicher voll ist oder die Escape-Taste gedrückt

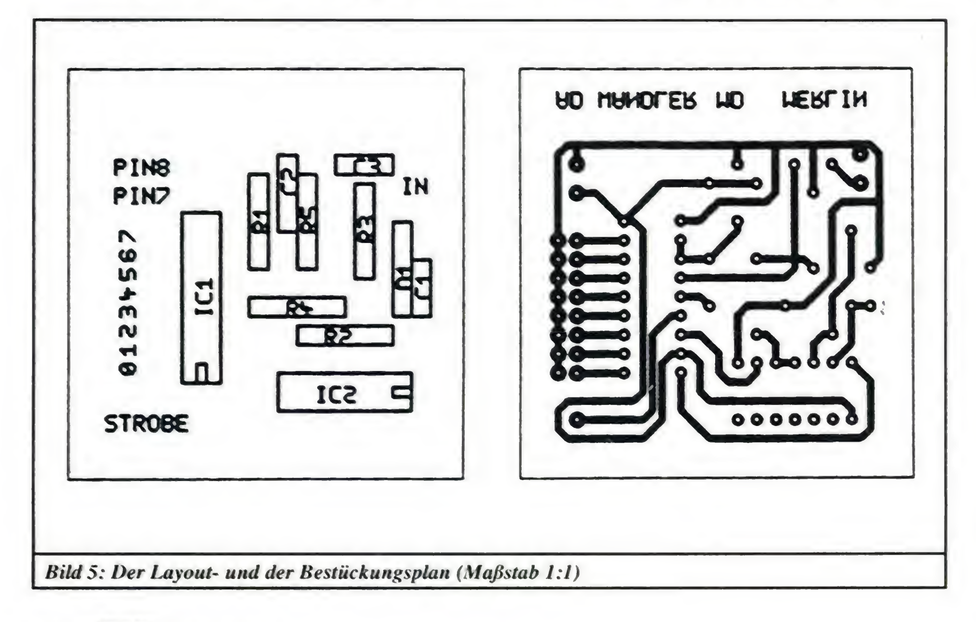


Bild 5: Der Layout- und der Bestückungsplan (Maßstab 1:1)

Stückliste

IC1:	ZN427
IC2:	14584
R1:	390Ω
R2:	22K
R3:	33K
R4:	82K
R5:	100K
D1:	1N4148
C1:	100pF
C2, C3:	10µF

1 D-Sub-Stecker (25-polig)
1 Chinchstecker (Kopfhörer)
1 Joystickbuchse (für ST)
1 IC-Sockel 14
1 IC-Sockel 18

wird. Das Stück wird dann im unteren Teil des Bildschirms als Graph dargestellt.

Mit F3 kann ein File von Diskette in den Speicher gelesen werden, um es dann abzuspielen. Es können übrigens auch Files geladen werden, die keine digitalisierte Musik enthalten. Es ist zwar relativ sinnlos, aber bestimmt interessant, wie sich das eine oder andere Programm 'anhört'.

Durch F4 können die im Speicher befindlichen Daten auf Diskette abgespeichert werden. Sollte auf einer Diskette weniger Platz sein, als für die Daten benötigt werden, wird das File einfach um den entsprechenden Teil gekürzt und dann auf Diskette abgespeichert. Ein paar Bytes werden auf der Diskette immer noch freigelassen.

Durch einen Druck auf F5 kann das Sample abgespielt werden. Es wird so lange wiederholt, bis die Space-Taste gedrückt wird.

Mit F6 kann das Stück rückwärts gespielt werden. Auch hier wird so lange wiederholt, bis die Space Taste gedrückt wird.

Mit F7 läßt sich die Aufnahme- bzw. Hörfrequenz verändern. Diese kann in dem Bereich von 3 bis 22 liegen. Nach dem Programmstart ist sie auf 20 gesetzt. Bei 3 ist die Wiedergabequalität am schlechtesten, bei 22 am besten. Dieser Wert beeinflußt die Menüpunkte 'Hören' und 'Aufnehmen'.

Mit der Funktionstaste F8 läßt sich die Abspielfrequenz verändern. Sie kann in einem Bereich von 3 bis 30 liegen. Dieser Wert beeinflußt die Abspielgeschwindigkeit der Menüpunkte Vorwärts- bzw. Rückwärtsspielen. 30 ist am schnellsten und 3 am langsamsten.

Über F9 läßt sich eine Diskette formatieren. Es gibt die Möglichkeit, entweder eine einseitige oder eine zweiseitige Diskette mit 10 Sektoren zu formatieren.

Mit F10 wird das Programm nach einer Sicherheitsabfrage verlassen.

Da das Programm sehr modular und Übersichtlich aufgebaut ist, dürfte es geübten Programmierern keine größere Schwierigkeiten bereiten, z.B. eine Funktion zur Definition von Blöcken oder andere, eigene Funktionen einzubauen.

Martin Backschat

```
 1: ;------------------------------
 2: ;  ST Sound Sampler 1.0
 3: ;  written 1988 by Martin Backschat, Bergstr.16,
      8071 Hepberg
 4: ;  mit IDEAL von OMIKRON ('z' zum Assemblieren und
 5: ;  'z,w sample.tos' zum Linken und Sichern)
 6: ;------------------------------
 7: start:
 8:   move.l  4(sp),a0     ; Basepage-Zeiger von Stack holen
 9:   move.l  $c(a0),d0    ; Länge des benötigten
                             Programmspeicher
10:   add.l   $14(a0),d0   ; aus Prg.-,Daten- und uninit.
                             Datenteillänge
11:   add.l   $1c(a0),d0   ; berechnen
12:   add.l   #$100,d0     ; + 256 Bytes Basepage
13:   move.l  d0,-(sp)     ; als Argument auf Stack
14:   pea     (a0)         ; Basis des Programmes
15:   pea     $004a0000    ; SETBLOCK
16:   trap    #1
17:   lea     12(sp),sp
18:   ;
19:   move.w  #4,-(sp)
20:   trap    #$e          ; Auflosung abfragen (0,1 Farbe
                             /2 Mono)
21:   addq.l  #2,sp
22:   subq.w  #1,d0
23:   bpl.s   nolowrez     ; wenn niedrige Auflosung in
                             mittlere schalten
24:   moveq.l #-1,d0
25:   move.w  #1,-(sp)
26:   move.l  d0,-(sp)     ; Bildschirmlage unverandert
                             lassen
27:   move.l  d0,-(sp)
28:   move.w  #5,-(sp)
29:   trap    #$e          ; Auflosung umschalten
30:   lea     12(sp),sp
31:   moveq.l #0,d0
32: nolowrez:
33:   move.w  d0,rez       ; 0 wenn Farbe 640x200, 1 wenn
                             Mono 640
        x400
34:   .dc.w   $a000        ; Grafikroutinen initialisieren
35:   move.l  a0,lineavar  ; Zeiger auf Grafikvariablen merken
36:   move.w  #1,24(a0)    ; Alle Grafiken in Rot bzw.
                             schwarz bei Mono
37:   moveq.l #-1,d0
38:   move.w  d0,34(a0)    ; Linienmuster ist durchzogene
                             Linie
39:   move.w  d0,32(a0)
40:   clr.w   36(a0)       ; Schreibmodus a.Ersetzen stellen
41:   ;
42:   clr.l   -(sp)
43:   move.w  #$20,-(sp)
44:   trap    #1           ; i.den Supervisormodus wechseln
45:   addq.l  #6,sp
46:   move.l  d0,oldsp     ; alten Stackzeiger merken
47:   ;
48:   moveq.l #-1,d0
49:   move.l  d0,-(sp)
50:   move.w  #$48,-(sp)
51:   trap    #1           ; Anzahl d.freien Bytes abfragen
52:   addq.l  #6,sp
53:   sub.l   #5*1024,d0   ; 5 KB dem System ubriglassen
54:   move.l  d0,freemem   ; Anzahl der verfugbaren Bytes
                             merken
55:   lea     memtext(pc),a0 ; u.d.Benutzer i.Text
                               einblenden
56:   move.l  #$ffff,d7
57:   move.l  d0,d1
58:   lsr.l   #5,d1
59:   lsr.l   #5,d1        ; /1024 daraus ergeben sich Anzahl KB
60:   move.l  #1000,d2     ; 4 stellige Zahl ausgeben
61:   moveq.l #3,d3
62: mainl:
63:   divu    d2,d1
64:   moveq.l #'0',d4      ; in ASCII-Zeichen umwandeln
65:   add.b   d1,d4
66:   move.b  d4,(a0)+     ; in Text schreiben
67:   swap    d1           ; Divisionsrest als neue Zahl
                             verwenden
68:   and.l   d7,d1        ; obere Langworthalfte loschen
                             (fur divu)
69:   divu    #10,d2       ; auch den Divisor anpassen
70:   dbf     d3,mainl     ; solange, bis Zahl komplett
                             ausgegeben
71:   move.l  d0,-(sp)
72:   move.w  #$48,-(sp)
73:   trap    #1           ; Verfugbaren Speicher reserv.
74:   addq.l  #6,sp
75:   move.l  d0,memadr    ; Adresse d.Speicherblocks merken
76:   ;
77:   clr.b   $484 ; Tastenwiederholung u.-klick abschalten
78:   pea     siditab(pc)  ; Sounddatentabelle
79:   move.w  #$20,-(sp)
80:   trap    #$e   ; Soundchip auf Digitalmusik vorbereiten
81:   addq.l  #6,sp
82:   ;
83:   lea     inittext(pc),a0
84:   bsr     print        ; Bildschirm loschen
85:   ;
86: menuloop:
87:   lea     menutext(pc),a0
88:   bsr     print        ; Programmenu ausgeben
89:   bsr     showgraf     ; Graphen des Samples anzeigen
90: menukey:
91:   bsr     getkey       ; Taste holen
92:   swap    d0           ; Funktionstasten nur uber
                             Scancodes erreichbar
93:   moveq.l #$3b,d1
94:   cmp.w   d1,d0        ; Taste zwischen F1 und F10?
95:   bmi.s   menukey
96:   cmp.w   #$44+1,d0
97:   bpl.s   menukey      ; nein, dann ignorieren
98:   ;
99:   sub.w   d1,d0        ; ansonsten aus Tastencode Offset
                             formen
```



STARKE SOFTWARE

ST ARCHIVAR
DIE ÜBERZEUGENDE DATEIVERWALTUNG

- Unterstützt die Produktion wissenschaftlicher Texte
- Verwaltet und bearbeitet Zitate u. Literaturangaben und Ausgabe als Text
- Komfortable Nutzung für Video- u. Adressendateien
- Datenverwaltung mit Dateien von bis zu 19 Datenfeldern (38 in Vorbereitung und 200 Datensätzen
- Dynamisch erweiterbar
- Veränderung der Dateimasken
- Speicherung der Daten als DIF oder SDF Datei, damit Übernahme in Datenbank möglich
- Beliebige Programme nachladbar u. auszuführen ohne ARCHIVAR zu verlassen (z. B. 1st Word)
- mit allen Textprogrammen kompatibel, die ASCII einlesen
- Ausdruck von Karten (z. B. Bibliothekskarten)
- Listen-Ausdruck auf Knopfdruck
- Etiketten-Ausdruck
- Einfache Druckerprogrammierung
- Einfach bedienbar mit der Maus

ST-ARCHIVAR enthält zahlreiche Hilfsprogramme

- Bibliotheks-Suchfunktion (ganze Disketten durchsuchen lassen)
- Uhr-Einstellung
- Info-Datei über alle ARCHIVAR-Dateien erstellen
- Druckersteuer-Programm
- Kopierprogramm
- Sortierprogramm
- Erzeugt RAM-DISK G bis 1750 KB
- Automatisches Kopieren in die RAM DISK G beim Start

und das alles für **DM 89,–**

ST-COMPUTER Index **NEU**
DAS JAHRESINHALTS-VERZEICHNIS 1986/1987

Wer sich einen schnellen Überblick darüber verschaffen möchte, welche Artikel 1986 und 1987 in **ST COMPUTER** erschienen sind, der kann jetzt mit **ST COMPUTER Index** die Jahrgänge 86/87 der Zeitschrift komfortabel als Handbuch und Nachschlagewerk nutzen.

Die Diskette enthält:

- **eine Indexdatei,** in die die Registereinteilung, zahlreiche Schlagwörter sowie Namen von Listings aufgenommen sind. Sie enthält 11 Datenfelder mit den Bezeichnungen Autor, Titel, Schlagwort, Stichwort, Index, Bemerkungen, Seitenangabe. Raum für eigene Einträge ist vorhanden.
- **ST_COMPX.PRG** zum Lesen, Suchen, Eingeben, Ausgeben. Es ist auf allen ST's in mittlerer und hoher Auflösung lauffähig.
- **ST_INDEX.EXE,** ein unter DOS lauffähiges Programm zum Lesen und Suchen der Dateieinträge.

ST_COMPX.PRG ist eine gekürzte Demoversion des Programmes ST_ARCHIVAR.

DM 29,–

ST AKTIE
AKTIENVERWALTUNG

- Aktien- u. Depotverwaltung u. Bilanzierung
- CHART-Analyse incl. Datenbank mit Kurven mit über 50 versch. Aktien beliebig aktualisierbar u. erweiterbar
- Sämtl. großen deutschen Standardwerte (55 St.) bis Okt. 87 fortgeführt
- Kto.-Führung für alle Orders u. andere Buchungen
- Kompakt, übersichtlich, Menü-orientiert, einfache Bedienung durch Maus, u. lfd. Programmhilfen, Graph. Darstellung standardgemäß, generiert selbständig

DM 69,–

ST BASIC-LIBS **NEU**
DIE GEM-SCHNITTSTELLE FÜR GFA-BASIC 2.0

Mit diesem Programm erstellen Sie vollständige GEM-Programme, ähnlich wie mit **GFA 3.0**, jedoch ohne **Compiler-Probleme**.

- Mehr als 200 Einzeldateien
- Vollständige AES-Bibliothek
- Die wichtigsten VDI-Routinen
- Komplette GEMDOS, BIOS- und XBIOS-Routinen
- Einfachste Erstellung von Dialogboxen durch leichte Einbindung der Ressource-Dateien
- Viele Hilfsprogramme
- Umfangreiches Beispielprogramm
- Ausführliche Programmbeschreibung

DM 29,–

ST LEARN **Neue Version**
DER VOKABELTRAINER FÜR DEN ST

programmiert v. D. Owerfeldt, Gewinner der GOLDENEN DISKETTE '87 für das beste Lernprogramm.

- Läuft vollständig unter GEM
- „Intelligente" Auswertung der Benutzereingaben
- Fehlerhäufigkeit einer Vokabel wird berücksichtigt
- Berücksichtigung mehrerer Bedeutungen eines Wortes
- Vielfältige Möglichkeiten des Lernens und der Abfrage
- Integriertes Lernspiel „HANGMAN"
- Spezielle Auswertung für unregelmäßige Verben (bei Eingabe von „to go" werden die anderen 2 Formen nachgefragt)
- Bei offensichtlicher Ähnlichkeit der Wörter wird wahlweise ein 2. Versuch zugelassen
- Trotz Einordnung der Vokabeln nach Lektionen oder Wissensgebieten ständig schneller Zugriff auf **alle** Vokabeln (nur durch Größe des Speichermediums begrenzt)
- Voller europäischer Zeichensatz (Zugriff durch die Maus unter GEM)
- Auch für Farbmonitor in mittlerer Auflösung
- Wörterbuchfunktion durchsucht alle Files eines Speichermediums nach einer Übersetzung ab
- Verbessertes Eingabeformular

ACHTUNG!!! NEU an ST-LEARN:

Allgemeines – Alle GEM-Routinen wurden überarbeitet u. optimiert, das Programm ist noch schneller geworden. – Besitzer eines SW-Monitors kommen in den Genuß einer neuen, etwas unkonventionellen (C) Mitteilung unter dem Menüpunkt **Über ST-LEARN.** – In der unteren Sonderzeichenleiste sind weitere Sonderzeichen hinzugekommen. – Im Lieferumfang befinden sich **jetzt auch mehrere englische Vokabeldateien, mit über 1.600 einfachen Grundwortschatzvokabeln,** unterteilt in 2 Schwierigkeitsstufen:

***EASY1.VOK** und **EASY2.VOK** enthalten je 400 einfachere Vokabeln.

***HARD1.VOK** und **HARD2.VOK** enthalten 800 etwas schwerere Vokabeln.

Weiterhin werden noch einige kleinere Vokabeldateien mit wichtigen Wörtern, etwa Strukturwörter oder Ordnungswörter mitgeliefert.

Eingabe- und Editierfeld – Hier ist der Knopf **'Löschen'** hinzugekommen, der es ermöglicht, die gerade angezeigte Vokabel zu löschen. Alle nachfolgenden Vokabeln werden aufgerückt.

Auswertung – Die Auswertung der Vokabeln wurde weiter verbessert, sie ist noch differenzierter geworden.

Druckeranpassung – Es ist jetzt möglich, **ST-LEARN** auf wirklich **jedem Drucker** anzupassen. Dazu gibt es im Ordner **LEARN_IT** das Konfigurationsprogramm PRINTER.

Einschaltmeldung – Hinzugekommen ist eine Einschaltmeldung, die direkt nach dem Starten des Programmes erscheint. Sie zeigt den freien Speicherplatz an.

Weitere Vorzüge von ST-LEARN – Vokabeldateien können auch nach Themen angelegt werden. Dennoch ist der Zugriff auf **alle** auf Diskette vorhandenen Vokabeln möglich. – Der **Status** einer Vokabel drückt deren Bekanntheitsgrad aus. Ist der Status z. B. kleiner als Null, so wurde dieser Ausdruck mindestens einmal nicht gekonnt etc. Je niedriger der Status, desto schlechter wurde die Vokabel nicht gekonnt und desto öfter wird sie vom Programm abgefragt.

DM 69,–

Update **DM 19,–**

TKC-HAUSHALT ST
ORDNUNG, ÜBERBLICK, PLANUNG – MIT DER HAUSHALTSBUCHFÜHRUNG FÜR ALLE

Leistungsmerkmale

- Monats- und Jahrsabschluß
- Benennung von bis zu 50 laufenden und 200 Kostenkonten
- Verwaltung von bis zu 50 Dauerbuchungen mit wählbarer Frequenz
- Bis zu 2.500 Buchungen pro Periode
- Volle Eingabekontrolle auf Fehler
- Erstellung von Bilanzen, Journals, Saldenlisten und Kontenrahmen
- Ausgabe von Buchungslisten mit Korrekturmöglichkeit
- Saldenbildung von Kontenblöcken
- Anpassung an jeden Drucker
- Schnittstelle zu anderen Programmen
- Voll in GEM eingebunden
- UP-DATE-SERVICE
  Monatsabschluß mit automatischer Saldenübertragung in die Datei für den Folgemonat
  Ausdruck aller wichtigen Daten
  Einsetzbar auf allen ATARI ST und MEGA ST mit S/W und Farbmonitor
  Handbuch

DM 129,–

TKC-EINNAHME/ ÜBERSCHUSS ST
DIE KOMFORTABLE BUCHFÜHRUNG FÜR FREIBERUFLER UND KLEINGEWERBETREIBENDE.

Ohne Vorkenntnisse leicht zu bedienen

Leistungsmerkmale

- Freie Wahl des Abschlußzeitraums (Periode)
- Voreinstellung von bis zu 5 Steuersätzen
- Einrichtung von bis zu 10 laufenden Konten
- Einrichtung von bis zu 200 Einnahme/ Ausgabe-Konten
- Erfassung von bis zu 2.500 Buchungen pro Periode
- Volle Eingabekontrolle auf Fehler (Löschen der letzten Buchung)
- Erstellung von Bilanzen
- Erstellung von Journalen
- Erstellung von Saldenlisten und Kontenrahmen
- Datenaufbereitung für USt.-Voranmeldung
- Ausdruck aller wichtigen Daten
- Anpassung an jeden Drucker
- Schnittstelle zu anderen Programmen
- Periodenabschluß mit automatischer Saldenübertragung in die Datei für die Folgeperiode
- Voll in GEM eingebunden
- Einsetzbar auf allen ATARI ST und MEGA ST mit S/W und Farbmonitor
- UP-DATE-SERVICE
  Ausgabe von Liquidität, Gewinn und Verlust
  Handbuch

DM 149,–

ST PLOT
KURVENDISKUSSIONS- U. FUNKTIONSPLOTTER-PRG.

- ist ein Programm, das mehr kann, als einfach nur Funktionen zeichnen. Es bestimmt die richtigen Ableitungen u. damit werden Null-, Extrem- und Wendestellen berechnet, es stellt den Definitionsbereich u. die Periode fest und vieles andere mehr.
- zeichnet die Schaubilder von bis zu 3 Funktionen u. deren erste u. zweite Ableitung gleichzeitig auf einen Bildschirm. Danach können Ausschnitte der Funktionen vergrößert und verkleinert werden.
- ist kpl. GEM- u. mausgesteuert. Die Bedienung ist somit sehr einfach. Sie brauchen nur die Funktion einzugeben, alle anderen Berechnungen erledigt das Programm.
- läuft sowohl in mittlerer wie auch in hoher Auflösung.
- ist nicht nur für Professoren, Studenten u. Schüler interessant. Es kann überall dort angewendet werden, wo man sich mit Funktionen beschäftigt.
- Umfangreiche Eingabemöglichkeiten, z. B. Verwendung sämtl. Funktionen eines wiss. Rechners. Defin. von 10 versch. Zahlenwerten als Konstanten, e u. Pi verwendbar.
- Komfortable Zeichnung, z. B. autom. Zeichnung u. Beschriftung der Achsen und deren Skalierung, Fehlerroutinen, damit keine unnötigen Asymptoten gezeichnet werden, beliebig viele Funktionen nacheinander in ein Bild einzeichenbar ermöglicht Funktionenvergleich u. Ablesung der Schnittpunkte etc.
- Beste Berechnung der Funktionswerte, z. B. Hohe Rechengeschwindigkeit, hohe Rechengenauigkeit.

DM 59,–

* alle Preise sind unverbindlich empfohlene Verkaufspreise

BESTELL-COUPON

an Heim-Verlag
Heidelberger Landstraße 194
6100 Darmstadt-Eberstadt

Bitte senden Sie mir: ____________________

____

zzgl. DM 5,- Versandkosten (unabhängig von bestellter Stückzahl)
☐ per Nachnahme ☐ Verrechnungsscheck liegt bei

Name, Vorname ____________________

Straße, Hausnr. ____________________

PLZ, Ort ____________________

Benutzen Sie auch die in ST COMPUTER vorhandene Bestellkarte.

Heim Verlag
Heidelberger Landstraße 194
6100 Darmstadt-Eberstadt
Telefon 06151-56057

```
100:    lsl.w   #2,d0
101:    lea     cmdtab(pc),a0 ; Tabelle mit den
                                Kommandoadressen
102:   move.l  0(a0,d0),a0 ; Adresse d.gewünschten Kommandos
        holen
103:    jsr     (a0)        ; anspringen
104:    ;
105:   bra.s   menuloop    ; wenn Kommando ausgeführt, dann
                              wieder ins Menü
106:
107: ; Gibt den Text, auf den A0 zeigt, aus
108: print:
109:    pea     (a0)
110:   move.w  #9,-(sp)
111:    trap    #1          ; PRINT TEXT
112:   addq.l  #6,sp
113:    rts
114:
115: ; Holt eine Taste (unteres D0.W  ist ASCII, oberes D0
      ist Scancode)
116: getkey:
117:   move.w  #7,-(sp)
118:    trap    #1
119:   addq.l  #2,sp
120:    rts
121:
122: ; Gibt eine Nachricht (Zeiger in A0) an den Anwender in
      Zeile 16 aus
123: printmsg:
124:    pea     (a0)            ; Zeiger merken
125:    lea     msg1text(pc),a0
126:    bsr     print           ; erstmal positionieren und
                                  Invers an
127:   move.l  (sp)+,a0
128:    bsr     print           ; jetzt Nachricht ausgeben
129:    lea     msg2text(pc),a0
130:    bra     print           ; und Invers ausschalten
131:
132: ; Löscht die Zeile 16, in der eine Nachricht steht
133: clrmsg:
134:    lea     clrmtext(pc),a0
135:    bra     print
136:
137: ; Berechnet in D0.L die  ASCII-Zahleneingabe (die in
      'in ptext' abgelegt ist)
138: calctext:
139:   lea     inptext(pc),a6 ; Hier steht die Zahleneingabe
140:   moveq.l #0,d0          ; Startwert 0
141: calcloop:
142:   moveq.l #0,d1
143:   move.b  (a6)+,d1 ; erstes Zeichen holen
144:   cmp.w   #'0',d1  ; ist Zeichen keine ASCII-Zahl, dann
145:   bmi.s   calcend  ; dies als Textende betrachten und
146:   cmp.w   #'9'+1,d1  ; stoppen
147:   bpl.s   calcend
148:   sub.b   #'0',d1  ; aus ASCII '0-9' die Zahl 0-9
                          berechnen
149:   mulu    #10,d0   ; Zahl um eine 10erpotenz gröPer
150:   add.w   d1,d0    ; Zahl zu Gesamtzahl addieren
151:   bra.s   calcloop
152: calcend:
153:    rts
154:
155: ; holt Eingabe von Benutzer; zuvor Nachricht ausgeben,
      wobei das erste Byte
156: ; des Textzeigers die maximale Eingabelänge angibt
157: gettext:
158:   moveq.l #0,d7
159:   move.l  d7,d6
160:   move.b  (a0)+,d7      ; Anzahl der erlaubten Eingaben
161:    pea     (a0)         ; Textzeiger merken
162:    lea     get1text(pc),a0
163:    bsr     print        ; erstmal positionieren
164:   move.l  (sp)+,a0
165:    bsr     print        ; Nachricht ausgeben
166:    lea     inptext(pc),a6  ; ab hier wird Eingabe
                                  gespeichert
167: keyinp1:
168:    bsr     getkey    ; Taste holen
169:    cmp.b   #$d,d0    ; wurde RETURN gedrückt?
170:    beq.s   endgett   ; ja, dann abbrechen
171:    cmp.b   #$8,d0    ; wurde BACKSPACE gedrückt?
172:    beq.s   delgett   ; ja, dann letztes Zeichen löschen
173:    cmp.b   d6,d7     ; ist erlaubte Textlänge
                            überschritten?
174:    beq.s   keyinp1   ; ja, dann Taste einfach ignorieren
175:   move.b  d0,(a6)+ ; ansonsten speichern
176:   addq.l  #1,d6    ; Zeichenzähler erhöhen
177:   move.w  d0,-(sp)
178:   move.w  #2,-(sp)
179:    trap    #1        ; u.Zeichen auf Bildschirm ausgeben
180:   addq.l  #4,sp
181:   bra.s   keyinp1   ; nächstes Zeichen einlesen
182: delgett:            ; Einsprung für BACKSPACE-Taste
183:   tst.b   d6        ; schon alle Zeichen gelöscht?
184:   beq.s   keyinp1   ; ja, dann ignorieren
185:   subq.l  #1,a6     ; ansonsten Zeichen a.Speicher
                          löschen
186:   subq.l  #1,d6     ; Zeichenzähler vermindern
187:    lea    deltext(pc),a0
188:    bsr    print     ; u.Zeichen auf Bildschirm löschen
189:   bra.s   keyinp1
190: endgett:            ; Einsprung für RETURN-Taste
191:   clr.b   (a6)      ; Eingabe mit $0-Kode beenden
192:    lea    get2text(pc),a0 ; blink.Cursor ausschalten
193:    bsr    print
194:    bra    clrmsg    ; Nachrichtenzeile löschen
195:
196: ; Gibt Fehlernachricht in A0 aus und wartet auf eine
      Taste
197: error:
198:    bsr    printmsg    ; Nachricht in A0 ausgeben
199:    bsr    getkey      ; auf Taste warten
200:    bra    clrmsg      ; Nachricht wieder löschen
201:
202:
203: ; Liest digitale Daten vom Printerport ein. Hierbei
204: ; mussen Parameter wie folgt übergeben werden:
205: ; D0.b = Scankode d.Taste, bei deren Druck frühzeitig
          abgebrochen werden soll
206: ; D1.w = Frq. in Hz (z.B. 22405  oder 8192); muss im
          Bereich 3000-22500 liegen
207: ; D2.b = Flag um zwischen Anhören (D2=0) und Aufnahme
          (D2=1) zu unterscheiden
208: ; A1 = Speicheradresse ab der die Daten beim Aufnehmen
         abgelegt werden sollen
209: ; A2 = bis zu dieser Adresse dürfen Daten abgelegt
         werden; wird diese Adresse
210: ;      erreicht, wird abgebrochen
211: input:
212:   move.l  a1,startadr ; Startadresse für später merken
213:   bsr     sndinit  ; Timer für Frq.syncronisation init.
214:   ;
215:   move.b  #$e,(a5)   ; ersten Strobe HIGH senden
216:   move.b  (a5),d7    ; damit wird Digitizer gestartet
217:   or.b    #$20,d7    ; Strobesignal in SID-Register 14
218:   move.b  d7,2(a5)
219:   move.b  #7,(a5)    ; Port B des Soundchips
220:   move.b  (a5),d7    ; auf Eingabe schalten
221:   and.b   #$7f,d7
222:   move.b  d7,2(a5)
223:   ;
224: inploop:
225:   moveq.l #0,d6
226:   move.b  #$f,(a5) ; Daten liegen an SID-Register 15 an
227:   move.b  (a5),d6    ; Daten auslesen
228:   ;
229:   move.b  #$e,(a5)
230:   move.b  (a5),d7
231:   and.b   #$df,d7    ; Strobe LOW ausgeben
232:   move.b  d7,2(a5)
233:   move.b  (a5),d7
234:   or.b    #$20,d7    ; Strobe HIGH ausgeben
235:   move.b  d7,2(a5)   ; um Digitizer zu <takten>
236:   ;
237:   tst.b   d2         ; Ausnehmen oder nur anhoren?
238:   beq.s   norec
239:   move.b  d6,(a1)+   ; über A1 speichern
240:   cmp.l   a1,a2      ; Endadresse schon erreicht?
241:   beq.s   inpend     ; ja, dann abbrechen
242: norec:
243:   and.b   #%11111100,d6 ; nur Bits 2-7 verwenden
244:   move.w  d6,d7      ; x3 nehmen um als Offset in
245:   add.w   d6,d7      ; Lautsprechertabelle zu dienen
246:   add.w   d6,d7
247:   movem.l 0(a3,d7),d5-d7 ; Lautsprecherdaten für alle 3
                               Kanäle
248:   movem.l d5-d7,(a5)  ; holen und in SID-Register
                             schreiben
249:   ;
250:   cmp.b   2(a4),d0   ; Abbruchtaste gedrückt?
251:   beq.s   inpend     ; ja, dann abbrechen
```

A–NET
DAS NETZWERK FÜR ATARI

Verbinden Sie Ihre Atari Computer störungsfrei mittels *modernster Lichtwellenleitertechnik*. Greifen Sie auf gemeinsam genutzte Daten und Ressourcen zu!
Und das alles mit der gewohnten Einfachheit der DESKTOP Operationen!

A–NET Grundeinheit 1650.–
ein Master- u. zwei Slave-interfaces, 15 m LWL, Software

FLASH ACCESS a.A.
Multiuser Datenbank für Softwareentwickler

PRIVATLIQUIDATION ST 525.–
Abrechnung nach GOÄ/GOZ, Patientenverwaltung, Mahnung, Kostenstellen.
(für Tierärzte in Vorbereitung)

LOHN & GEHALT ST 725.–
GEM unterstützte Bruttolohn- und Nettolohnabrechnung, Auswertung, Formulare, Überweisungen.

DM COMPUTER

DM COMPUTER GMBH
Durlacherstraße 39
7530 Pforzheim
Tel.:07231-13939
Tlx.:783248

Funkcenter Mitte GmbH

Klosterstr. 130 · 4000 Düsseldorf 1
Tel. 02 11 / 36 25 22 · FAX 02 11 / 3 60 19

Produkt	Preis
GFA Basic 2.02 Interpreter	79,–
GFA Basic Compiler	79,–
GFA Basic 3.0 Interpreter	168,–
GFA Draft Plus	298,–
GFA Draft	168,–
GFA Objekt	168,–
GFA Vektor	79,–
GFA GEM-Starter	55,–
GFA Movie	128,–
GFA Artist	128,–
GFA Farbkonverter	55,–
GFA Monokonverter	55,–
GFA Floppyspeeder	55,–
GFA Basic 68881	298,–
GFA Assembler	168,–
GFA Raytrace	128,–
COPY II ST (Central Point Inc.)	88,–

über 300 Public Domain Disketten für ATARI!

Katalogdiskette gegen 5,– Briefmarken oder Schein anfordern.

MAILBOX 24 Std. ONLINE
02 11 / 36 01 04 8,N,1



Das seit über 2 Jahren bewährte Kopierprogramm copySTar gibt es nun in einer völlig neu überarbeiteten und optimierten Version.

Vollautomatische Kopierschutzerkennung, auch bei "harten Fällen". Daher sind **keine Parameter** Eingaben **nötig**.

Kopiert **alle** ST Disketten, **normale** und **kopiergeschützte** ohne zusätzliche teure Hardware.

Superschnelle Kopien von "normalen" Disketten unter 30 Sekunden. Das Format der Disketten (9,10,11 Sektoren/SS-DS) wird dabei automatisch erkannt und richtig kopiert.

Automatische Fehlerkorrektur während des Kopiervorganges. Die Kopien von teilweise **defekten Disketten sind** oft **wieder** vollständig lauffähig und **fehlerfrei**.

Eine **umfangreiche Statusanzeige** beim Kopieren und Formatieren zeigt Ihnen genau, auf welchem Track/Sektor es Probleme oder Fehler beim Schreiben oder Lesen von Daten gegeben hat.

Superschnelle Formatierung von Disketten (9,10,11 Sektoren). Erweiterung der Disk-Kapazität auf über 912KB.

Konvertiert in **Spezialformat für doppelte Geschwindigkeit** ohne zusätzliche Hardware.

Qualitäts-**Prüfung von Leerdisketten**.

Drehzahl-Test des Disk-Laufwerkes. (Eine falsche Drehzahl kann die Ursache für viele Schreib-/Lesefehler sein.)

Super-Update-Service! GEM-Bedienung. **Der Preis DM 169,-**

Besitzer eines älteren copySTar erhalten die neue Version gegen eine geringe Gebühr. Bitte rufen Sie uns an.

T.L.D.U. The Last Disc Utility

Die komplette Disk-Utility mit wirklich allen Funktionen. Von vielen Fachzeitschriften für **SEHR GUT** befunden. Arbeitet auch mit Hard-und Ramdisk.

Voll programmierbar durch ·C· ähnliche Macrosprache.

Echte (Sub)-Directory und Fat Funktionen mit Grafik.

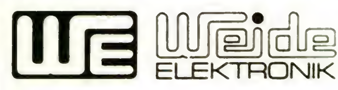
10 Buffer, **Disassembler,** Folgemodus, Suchen, Vergleichen, Drucken, **große Makro-Bibliothek** u.s.w. Ein-/Ausgabe in ASCII, EBCDIC, HEX oder DEZIMAL möglich. **nur DM 149,-**

Detective v1.0

Erstellt editierbare **Assemblersource-Dateien** von jedem ST-Programm. Verwaltet Text-,Data-,BSS-Bereiche, **Symbole**, Marker und mehr. Ein Muß für jeden Programmierer. **nur DM 149,-**

Wichtig!! Alle Programme enthalten ein **ausführliches Handbuch** in Deutsch und **keinen Kopierschutz**.

Wo? Natürlich bei Ihrem Atari-Händler!
oder direkt bei
STARSOFT
Peinerstr. 81 D-3000 Hannover 81
Tel. 0511/837 99 77



Weide Elektronik GmbH
Regerstraße 34
D-4010 Hilden
Telefon (0 21 03) 4 12 26

Ladenlokal: Gustav-Mahler-Straße im Einkaufszentrum
Telefon Ladenlokal (0 21 03) 3 18 80 · Telefax (0 21 03) 3 18 20
viele Parkplätze direkt am Laden

ATARI System-fachhändler

Leasing für Komplettsysteme möglich
Atari + PC

Schweiz
SENN Computer AG
Langstrasse 31
CH-8021 Zürich
Tel. (01) 2 41 73 73

Niederland
COMMEDIA
1e Looiersdwarsstr. 12
1016 VM Amsterdam
Tel. (0 20) 23 17 40

ATARI steckbar SPEICHERKARTEN

auf **1 MB**yte für 260/520 STM ... a. A.
auf **2 MB**yte für 520 ST+/1040 ST ... a. A.
(6 Lötpunkte)
auf **2,5 MB/4 MB** ... a. A.

Jede Erweiterung einzeln im Rechner getestet! Sehr einfacher Einbau ohne Löten. Gut bebilderte Einbauanleitung. Vergoldete Mikro-Steckkontakte, dadurch optimale Schonung des MMU-Sockels.
Achten Sie auf Mikro-Steckkontakte! Kein Bildschirmflimmern. Keine zus. Software. Ohne zus. Stromversorgung.

Echtzeituhr PLUS ... DM 129,–
Keine Software nötig
Jede Uhr im Rechner getestet und gestellt. Interner Einbau ohne Löten. Dadurch freier ROM-PORT. Immer aktuelle Zeit und aktuelles Datum. Keine Software nötig. Hohe Genauigkeit. Schaltjahrerkennung.

VIDEO SOUND BOX ... DM 248,–
Ihr ST am Fernseher. Klangkräftige 3-Wege-Box mit integriertem HF-Modulator zum Direktanschluß aller ATARI ST an den Fernseher. Unübertroffene Bildqualität. Super Sound!

EPROMKARTE 64 KB .... DM 12,90
mit vergoldeter Kontaktleiste für alle ATARI ST

Computer (520ST, 1040ST) ..... a. A.

DRUCKER
NEC P6 plus ... DM 1.598,–
Star LC 10, Laserdrucker a. A. auf Lager.

NLQ ... NLQ ... NLQ ... NLQ ... NLQ
Aufrüstsatz für alle EPSON MX, RX, FX, JX Drucker, Apple Macintosh Drucker Emulation (FX&JX). Viele Features! INFO anfordern.
FX DM 149,– MX DM 129,– RX DM 98,–

LAUFWERKE für ATARI ST

3,5" Einzellaufwerk ... DM 289,–
5¼" Einzellaufwerk ... DM 498,–

ICD HARDDISKS
(ST-HDXX+)

- **1 herausgeführter SCSI-Port** zum Anschluß von Standard PC-Peripherie; unterstützt werden bis zu 8 SCSI Geräte
- **1 x DMA Eingang, 1 x DMA Ausgang**
- Daisy Chaining am DMA Port möglich.
- **extrem leiser Lüfter**, minimale Laufgeräusche
- **100 % kompatibel** mit Atari Harddisks
- Alle Harddisks laufen auch mit Turbo DOS
- Eingebaute **Echtzeit-Uhr**
- Treiber führt speziellen Verify nach Schreib/Lesevorgängen durch
- Bei 50 MB und 100 MB Harddisks Autopark und Autoheadlock Mechanismus

84,9 MB form 2.698,–
Platten von 20 – 100 MB

TRAKBALL statt Maus ... DM 99,–
Ersatzteile, Zubehör ... a. A.
Floppystecker ... DM 8,90
Monitorbuchse ... DM 8,90
Monitorstecker ... DM 8,90
Floppykabel ... DM 19,90

ST HOST ADAPTER
(ST-HOAD)

- erlaubt **Anschluß von IBM kompatiblen Harddisks** (Typ ST506/ST412) an Ihren Atari ST
- unterstützt bis zu **8 SCSI Geräte** am DMA Port
- **1 x DMA Eingang, 1 x DMA Ausgang, 1 SCSI Port**
- **eingebaute Echtzeit-Uhr**
- enthält vollständige Softwareunterstützung (Treiber, Formatierungs- u. Partitionierungssoftware f. div. Harddrives, z. B.: Rodtime A, Seagate, Xebec 1440, WD1002-SHD, WD1002S-SHD)
- Treiber führt speziellen Verify nach jeder Schreib/Leseoperation durch; dadurch **hohe Datensicherheit**
- spezielles Programm, welches bei Kopiervorgängen die abgespeicherte Zeit unverändert läßt im Lieferumfang enthalten.

ST Host Adapter ... DM 348,–
ST Host Adapter mit Adaptec 400A MFM ... DM 648,–
ST Host Adapter mit Adaptec 4070 RLL ... DM 748,–

Produkt	Preis
GFA 3.0 Interpreter	189,–
Pro Fortran	378,–
Pro Pascal	248,–
OF-9 (Prof.)	1598,–
Signum II	398,–
GFA-DRAFT PLUS	329,–
Omikron Basic Interp.	178,–
BS-Handel	498,–
BS-Fibu	548,–

Coprozessor 68881 ... DM 890,–

In Ihrem ATARI ST. Mit Software für Megamax C, Mark Williams C, DRI C, Lattice C, Prospero Fortran 77, Modula II, CCD Pascal + AC-Fortran erhöht die Rechengeschwindigkeit z. T. um Faktor 900. Einfachster Einbau, rein steckbar – ohne Löten. Unbedingt ausführl. INFO anfordern!

68881
noch viel schneller mit
WCL
(Weide Coprocessor Language)

WARUM WCL?

Bei Hochsprachen Compilern werden die Daten in den Coprozessor hineingeschoben, dort berechnet und wieder herausgeholt ohne Rücksicht darauf, ob diese Daten eventuell direkt im 68881 weiterverarbeitet werden können. WCL ermöglicht es auf einfache Art und Weise Funktionen und Prozeduren zu erstellen, deren Zwischenergebnisse weitestgehend in den Coprozessorregistern gehalten werden.

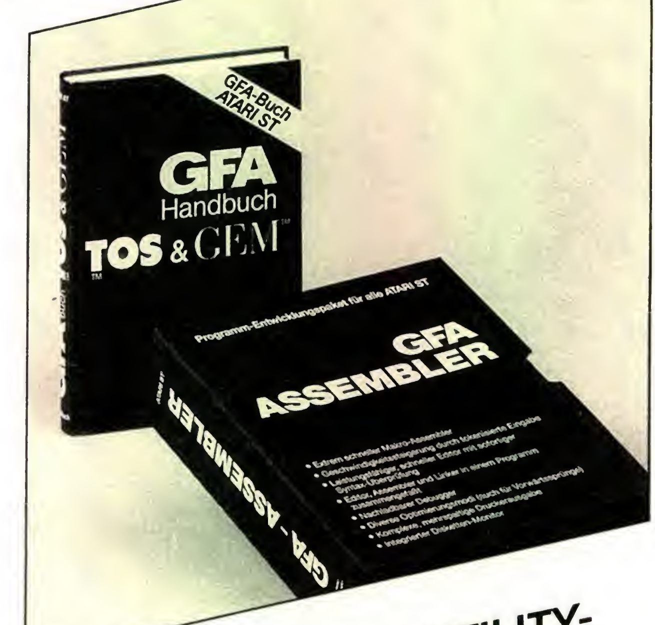
Softwareentwickler aufgepaßt!!!
WCL ist auch als Tool lieferbar. Gibt u. a. linearen, kommentierten 68000 Assemblersource aus! Enorme Zeitersparnis bei der Programmentwicklung!

Scanner HAWK 432 für ATARI ST/Amiga/PC Flachbettscanner mit CCD-Sensor. 64 Graustufen. Auflösung 400 DPI. Ist auch als Kopierer und Drucker einsetzbar. Für DTP einsetzbar. Bildformat für Publishing Partner, Fleetstreet Publisher, Monostar plus, Stad, Word + Degas, Profi Painter, Druckertreiber für NEC P6/7, STAR NL10, Canon LPBB u.v.m. Telefax wird demnächst möglich sein! Software zur Schriftenerkennung ist lieferbar. Unbedingt INFO anfordern.

Komplettes DTP System bestehend aus Rechner, Monitor, Harddisk, Scanner, Calamus, DTP oder Timeworks DTP a. A.
Privatliquidation incl. Unfallabrechnung für **Ärzte** DM 998,–
Komplette EDV Auftragsabwicklung a. A. Komplette EDV Buchhaltung a. A. Einarbeitung und Installation möglich!
Über alle Produkte auch INFO's erhältlich. Alle Preise zuzüglich Verpackung und Versand. Händleranfragen erwünscht.

```
252: inpsync:
253:   btst    #4,$d(a6)   ; solange warten, bis TIMER D
254:   beq.s   inpsync ; runtergezählt hat (Syncronisation!)
255:   bclr    #4,$d(a6)   ; TIMER D zurücksetzen
256:   bra     inploop     ; und nächste Daten einlesen
257: inpend:               ; Einsprung für Abbrechen
258:   tst.b   d2          ; wurde aufgenommen?
259:   beq.s   inpend2
260:   sub.l   startadr,a1 ; ja, dann die Länge des
                            aufgenommenen
261:   move.l  a1,smpllen  ; Bereiches festhalten
262: inpend2:
263:   bra     sndinit2    ; TIMER D wieder zurücksetzen
264:
265:
266: ; Spielt digitale Daten im Speicher ab. Parameter
     werden wie folgt übergeben:
267: ; D0.b = Scankode der Taste, bei deren Druck spielen
           abgebrochen werden soll
268: ; D1.w = Frq. in Hz (z.B. 22405 oder 8192); muss im
           Bereich 3000-30000 liegen
269: ; D2.b = Flag zur Unterscheidung ob vorwärts (D2=0)
270: ;        oder rückwärts (D2=1) gespielt werden soll
271: ; D3.w = Anzahl der Wiederholungen
272: ; A1 = Speicheradresse ab der d.digitalen Daten liegen
273: ; A2 = Endadresse der digitalen Daten
274: play:
275:   subq.w #1,d3 ; Anzahl Wiederholungen f.DBF-Schleife
276:   move.l a1,startadr ; Start- und Endadresse für
277:   move.l a2,endadr   ; Wiederholung merken
278:   bsr    sndinit     ; TIMER D für Syncronisation init.
279:   moveq.l #0,d4
280: playloop:
281:   tst.b  d2          ; vorwärts oder rückwärts spielen?
282:   bne.s  playb
283:   move.b (a1)+,d4 ; vorwärts, über Startadresse A1
284:   cmp.l  a1,a2    ; Daten lesen; Endadresse erreicht?
285:   bne.s  playf1   ; nein...
286:   dbf    d3,playfla ; wenn ja, Wiederholungszähler
287:   bra.s  playend  ; vermindern und bei -1 abbrechen
288: playfla:
289:   move.l startadr,a1 ; Startadresse A1 wieder
                           zurücksetzen
290: playf1:
291:   bra.s  play1
292: playb:               ; Einsprung fürs Rückwärtsspielen
293:   move.b -(a2),d4 ; über Endadresse A2 Daten lesen
294:   cmp.l  a1,a2    ; Startadresse schon erreicht?
295:   bne.s  play1    ; nein...ok
296:   dbf    d3,play1a ; wenn ja, dann Wiederholungszähler
297:   bra.s  playend  ; vermindern und bei -1 abbrechen
298: play1a:
299:   move.l endadr,a2 ; Endadresse A2 wieder zurücksetzen
300: play1:
301:   and.b  #%11111100,d4 ; nur Bits 2-7 verwenden
302:   move.w d4,d7     ; x3 um als Offset in
303:   add.w  d4,d7     ; Lautsprechertabelle zu dienen
304:   add.w  d4,d7
305:   movem.l 0(a3,d7),d5-d7 ; Lautsprecherdaten holen und
306:   movem.l d5-d7,(a5) ; in Lautsprecher d.SID schreiben
307:   ;
308:   cmp.b  2(a4),d0  ; Abbruchtaste gedruckt?
309:   beq.s  playend   ; ja, dann abbrechen
310: playsync:
311:   btst   #4,$d(a6) ; warten, bis TIMER D runtergezählt
312:   beq.s  playsync  ; hat (Syncronisation!)
313:   bclr   #4,$d(a6) ; TIMER D zurücksetzen
314:   bra    playloop  ; und weitermachen
315: playend:           ; Einsprung fürs Abbrechen
316:   bra    sndinit2  ; TIMER D zurücksetzen
317:
318: ; Initialisiert TIMER D und bereitet System auf
     Abspielen/Aufnehmen vor
319: sndinit:
320:   move.w #$2700,sr    ; ab jetzt kein IRQ mehr!
321:   lea    voldat(pc),a3 ; Lautsprechertabellenbasis
                              in A3
322:   move.w #$fc00,a4    ; Tastatur-ACIA-Basis in A4
323:   move.w #$8800,a5    ; SID-Basis in A5
324:   move.w #$fa00,a6    ; MFP-Basis in A6
325:   ;
326:   move.l #2457600,d7  ; aus Hz-Zahl in D1.w Timerdaten
327:   divu   d1,d7        ; berechnen mit Formel
328:   lsr.w  #2,d7        ; TI-DATA x 4 = TAKT / HZ
329:   move.b d7,$25(a6)   ; Timerstartwert
330:   move.b $1d(a6),d7   ; Vorteiler des TIMERs D
331:   and.b  #$f0,d7      ; ist 4
332:   or.b   #$01,d7
333:   move.b d7,$1d(a6)
334:   bset   #4,9(a6)     ; TIMER D muss IRQ signalisieren
335:   rts                 ; können
336:
337: ; TIMER D zurücksetzen und System wieder normalisieren
338: sndinit2:
339:   move.b $1d(a6),d7
340:   and.b  #$f0,d7      ; TIMER D stoppen
341:   move.b d7,$1d(a6)
342:   bclr   #4,9(a6)
343:   move.w #$2300,sr    ; IRQs wieder erlauben
344:   rts
345:
346: ; Beendet nach Nachfrage das Programm
347: quit:
348:   lea    quittext(pc),a0
349:   bsr    printmsg  ; Nachfragetext ausgeben
350: quitkey:
351:   bsr    getkey    ; Taste holen
352:   cmp.b  #'j',d0   ; wenn j für JA gedrückt, dann
353:   beq.s  surequit  ; abbrechen
354:   cmp.b  #'J',d0   ; wenn J für JA gedrückt, dann
355:   beq.s  surequit  ; abbrechen, ansonsten wieder ins
356:   bra    clrmsg    ; Menü zurückkehren
357: surequit:
358:   addq.l #4,sp ; Rücksprungadr d.JSR vom SP entfernen
359:   move.b #7,$484 ; Tastenwiederholung u.Klick einsch.
360:   move.l oldsp,-(sp)
361:   move.w #$20,-(sp)
362:   trap   #1        ; wieder in den USER-Modus zuruck
363:   addq.l #6,sp
364:   move.l memadr,-(sp) ; Basis d.reservierten Speichers
365:   move.w #$49,-(sp)
366:   trap   #1      ; belegten Speicher wieder freigeben
367:   addq.l #6,sp
368:   clr.w  -(sp)
369:   trap   #1      ; Programm beenden
370:
371: ; Digital Sound vom Druckerport anhören
372: listen:
373:   lea    heartext(pc),a0
374: listen1:
375:   bsr    printmsg  ; Nachricht ausgeben
376:   moveq.l #$39,d0  ; Abbruchtaste ist SPACE
377:   move.w recfrq,d1 ; z.Anhören Aufnahmefrq. verwenden
378:   moveq.l #0,d2    ; Flag auf Anhören setzen
379:   bsr    input     ; Anhören
380:   bra    clrmsg    ; Nachricht löschen u.i.Menu zurück
381:
382: ; Digital Sound vom Druckerport aufnehmen
383: record:
384:   lea    rec1text(pc),a0
385:   bsr    listen1     ; erstmal Anhoren und bei SPACE
386:   lea    rec2text(pc),a0 ; Aufnahme starten
387:   bsr    printmsg
388:   moveq.l #$01,d0        ; bei Aufnahme ist ESC die
                               Abbruchtaste
389:   move.w recfrq,d1       ; Aufnahmefrq.
390:   moveq.l #1,d2          ; Flag für Aufnehmen
391:   move.l memadr,a1       ; Startadresse, ab der
                               aufgenommen wird
392:   move.l a1,a2
393:   add.l  freemem,a2      ; Endadresse d.Speicherblocks
394:   bsr    input           ; Aufnehmen
395:   bra    clrmsg      ; Nachricht löschen und ins Menü
396:
397: ; Abspielfrq. neu setzen (Eingabe des Anwenders in KHz)
398: setplay:
399:   lea    setptext(pc),a0 ; Eingabetext
400:   move.l #playfrq,frqptr ; Zeiger a.d.Abspielfrq (.w)
401:   moveq.l #30,d0         ; maximale Frq. ist 30 KHz
402:   bra.s  setlabl
403: setrec:
404:   lea    setrtext(pc),a0 ; Eingabetext
405:   move.l #recfrq,frqptr  ; Zeiger a.d.Anfnamhefrq.(.w)
406:   moveq.l #22,d0         ; maximale Frq. ist 22 KHz
407: setlabl:
408:   move.w d0,-(sp)   ; max. Frq. merken
409:   bsr    gettext    ; erstmal Benutzereingabe holen
410:   bsr    calctext   ; Eingabe i.Zahl in D0.W umwandeln
411:   move.w (sp)+,d1   ; max. Frq. zurückholen
412:   cmp.w  #3,d0      ; min Frq. 3 KHz unterschritten
413:   blt.s  seterr     ; oder maximale Frq. 22/30 KHz
414:   cmp.w  d1,d0      ; überschritten, dann
```

Software-Leistung



Für Profi-Programmierer

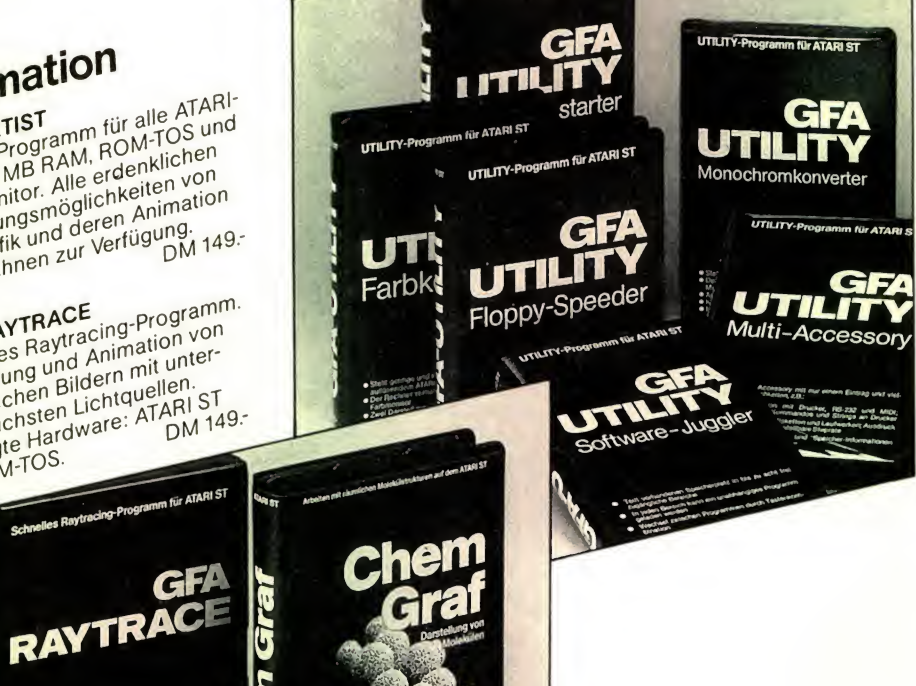
GFA-ASSEMBLER
Extrem schnelles Programm-Entwicklungspaket für alle ATARI ST. Editor, Assembler und Linker sind in einem Programm zusammengefaßt, der Debugger ist nachladbar.
DM 149.-

GFA-Handbuch TOS & GEM
Das Nachschlagewerk für jeden ST-Programmierer. Das Buch enthält eine komplette Übersicht sämtlicher Routinen und dokumentierter Variablen des Betriebssystems.
ISBN 3-89317-002-2
368 Seiten + 32 Seiten Indexheft
DM 49.-

Animation

GFA-ARTIST
Design-Programm für alle ATARI-ST mit 1 MB RAM, ROM-TOS und Farbmonitor. Alle erdenklichen Gestaltungsmöglichkeiten von Farbgrafik und deren Animation stehen Ihnen zur Verfügung.
DM 149.-

GFA-RAYTRACE
Schnelles Raytracing-Programm. Darstellung und Animation von realistischen Bildern mit unterschiedlichsten Lichtquellen. Benötigte Hardware: ATARI ST mit ROM-TOS.
DM 149.-



ChemGraf
Darstellung von Molekülen und deren Bindungen. Das Siegerprogramm aus dem Programmierwettbewerb der GFA und Markt & Technik.
DM 79.-

GFA-UTILITY-Serie

Farbkonverter
Stellt die geringe und mittlere Auflösung auf dem hochauflösenden Monitor dar. Konvertiert die meisten Farb-Anwender-Programme. Spiele werden in der Regel nicht konvertiert.
DM 59.-

Monochrom-Konverter
Die hohe Auflösung ist mit diesem Utility auf dem Farbmonitor darstellbar. Die meisten Monochrom-Programme werden konvertiert.
DM 59.-

Floppy-Speeder
Diskettenoperationen werden bis zu Faktor 20 beschleunigt. Die RESET-feste GFA-RAM-Disk ist im Lieferumfang enthalten.
DM 59.-

GEM-Autostarter
Bis zu 10 Programme können hintereinander aus dem Auto-Ordner gestartet werden. Die Übergabe von Parametern ist möglich.
DM 59.-

Software-Juggler
Teilt vorhandenen Speicherplatz in bis zu acht frei zugängliche Bereiche. In jeden Bereich kann ein unabhängiges Programm geladen werden.
DM 59.-

Multi-Accessory
Universelles Accessory mit nur einem Eintrag, jedoch vielseitigen Möglichkeiten für die tägliche Arbeit mit dem ATARI ST.
DM 59.-

GFA Systemtechnik GmbH
Heerdter Sandberg 30
D-4000 Düsseldorf 11
Telefon 02 11/58 80 11



```
415:   bhi.s   seterr     ; Fehlermeldung ausgeben
416:   mulu    #1000,d0   ; aus KHz Hz machen (x1000)
417:   move.l  frqptr,a0  ; Zeiger auf Frequenz (.w)
418:   move.w  d0,(a0)    ; Hz-Zahl speichern
419:   rts                ; und zurück ins Menü
420: seterr:
421:   lea     setetext(pc),a0 ; Fehlermeldung ausgeben
422:   bra     error
423:
424: ; digitalen Sound im Speicher rückwärtsspielen
425: playback:
426:   moveq.l #1,d2      ; Flag für rückwärts
427:   bra.s   playsnd
428: ; digitalen Sound im Speicher abspielen
429: playforw:
430:   moveq.l #0,d2      ; Flag für vorwärts
431: playsnd:
432:   move.w  d2,-(sp)        ; Flag merken
433:   lea     playtext(pc),a0 ; Nachricht ausgeben
434:   bsr     printmsg
435:   moveq.l #$39,d0    ; Abbruchtaste ist SPACE
436:   move.w  playfrq,d1 ; Abspielfrquenz holen
437:   move.w  (sp)+,d2   ; Flag vor/rückwärts
438:   moveq.l #-1,d3     ; Anzahl d.Wiederhol. unendlich
439:   move.l  memadr,a1  ; Startadresse des Sounds
440:   move.l  a1,a2
441:   add.l   smpllen,a2 ; Endadresse des
                           Sound-Speicherblocks
442:   bsr     play       ; und abspielen
443:   bra     clrmsg  ; Nachricht löschen u.i.Menü zurück
444:
445: ; setzt  das Laufwerk, wenn bei Filenameneingabe
            Laufwerksangabe vorkommt (A:...)
446: setdrv:
447:   cmp.b   #':',inptext+1 ; kommt Laufwerksangabe vor?
448:   bne.s   setdrve        ; nein, dann zurück
449:   moveq.l #0,d0
450:   move.b  inptext,d0 ; Laufwerksnummer (A,B,C...) holen
451:   and.b   #%11,d0        ; für Funktions in Nummern
(0,1,2)
452:   subq.w  #1,d0          ; umformen
453:   move.w  d0,-(sp)
454:   move.w  #$e,-(sp)
455:   trap    #1             ; SET DRIVE
456:   addq.l  #4,sp
457: setdrve:
458:   rts
459:
460: ; Lädt eine Sample-Datei von Disk
461: load:
462:   lea     loadtext(pc),a0
463:   bsr     gettext      ; Dateinamen holen
464:   bsr     setdrv       ; ggf. Laufwerk setzen
465:   clr.w   -(sp)
466:   pea     inptext(pc)
467:   move.w  #$3d,-(sp)
468:   trap    #1           ; Datei öffnen
469:   addq.l  #8,sp
470:   tst.l   d0           ; Fehler beim Öffnen?
471:   bmi.s   loaderr      ; ja, dann Datei nicht gefunden..
472:   move.w  d0,-(sp)
473:   move.l  memadr,-(sp)  ; Startadresse
474:   move.l  freemem,-(sp) ; Länge des Speicherblocks
475:   move.w  d0,-(sp)
476:   move.w  #$3f,-(sp)
477:   trap    #1            ; Datei einlesen
478:   lea     12(sp),sp
479:   move.l  d0,smpllen    ; Anzahl der eingelesene Bytes
480:   move.w  #$3e,-(sp)    ; als Sample-Länge übernehmen
481:   trap    #1            ; Datei wieder schliessen
482:   addq.l  #4,sp
483:   rts
484: loaderr:
485:   lea     loadetxt(pc),a0  ; <File not found>-
                                  Fehlermeldung
486:   bra     error          ; ausgeben
487:
488: ; Speichert aktuelles Sample auf Disk
489: save:
490:   lea     savetext(pc),a0
491:   bsr     gettext        ; Dateiname holen
492:   bsr     setdrv         ; ggf. Laufwerk setzen
493:   clr.w   -(sp)
494:   pea     diskinfo(pc)
495:   move.w  #$36,-(sp)
496:   trap    #1             ; GET DISKINFO
497:   addq.l  #8,sp
498:   move.l  smpllen,d7    ; Länge des Samples
499:   move.l  diskinfo,d0   ; freie Clusters auf Disk
500:   subq.l  #2,d0
501:   mulu    #1024,d0      ; -> freie Bytes auf Disk
502:   cmp.l   d7,d0         ; zu wenig freie Bytes auf Disk?
503:   bmi.s   saveerr       ; ja, dann Sample einfach kürzen
504: saveerr1:
505:   clr.w   -(sp)
506:   pea     inptext(pc)
507:   move.w  #$3c,-(sp)
508:   trap    #1            ; Datei auf Disk eröffnen
509:   addq.l  #8,sp
510:   move.w  d0,-(sp)
511:   move.l  memadr,-(sp)  ; Startadresse des Samples
512:   move.l  d7,-(sp)      ; (angepasste) Länge des Samples
513:   move.w  d0,-(sp)
514:   move.w  #$40,-(sp)
515:   trap    #1            ; Sample auf Diskette speichern
516:   lea     12(sp),sp
517:   move.w  #$3e,-(sp)
518:   trap    #1            ; Datei schliessen
519:   addq.l  #4,sp
520:   bra     clrmsg        ; und zurück ins Menü
521: saveerr: ; Einsprung, wenn Diskkapazität zu gering
522:   exg     d0,d7         ; Samplelänge einfach anpassen
523:   lea     saveetxt(pc),a0 ; Benutzermitteilung
                                 ü.Anpassung
524:   bsr     printmsg        ; geben
525:   bra     saveerr1        ; und speichern...
526:
527: ; formatiert eine Diskette 1- oder 2-seitig mit
   80 Tracks/10 Sektoren
528: format:
529:   lea     formtext(pc),a0
530:   bsr     gettext   ; Anzahl d.Seiten vom Benutzer holen
531:   bsr     calctext  ; in Zahl umwandeln (1 oder 2)
532:   move.l  d0,d7
533:   subq.l  #1,d7     ; 1/2 in 0/1 formen
534:   bmi     formend   ; bei unzulässiger Eingabe abbrechen
535:   cmp.w   #2,d7
536:   bpl     formend
537:   ;
538:   lea     formtpuf(pc),a6 ; Datenpuffer fürs Formatieren
539:   moveq.l #79,d6          ; beginnen mit Track 80
540: nexttr:                   ; Label für Trackzähler
541:   move.w  d7,d5
542: nextside: ; Label für Seitenzähler
543:   clr.w   -(sp)
544:   move.l  #$87654321,-(sp)
545:   move.w  #1,-(sp)
546:   move.w  d5,-(sp)
547:   move.w  d6,-(sp)
548:   move.w  #10,-(sp)
549:   clr.w   -(sp)
550:   clr.l   -(sp)
551:   pea     (a6)
552:   move.w  #10,-(sp)
553:   trap    #$e        ; Track D6/Seite D5 formatieren
554:   lea     26(sp),sp
555:   dbf     d5,nextside ; Seitenzähler
556:   dbf     d6,nexttr   ; Trackzähler
557:   ;
558:   bsr     clrpuf      ; Puffer löschen
559:   clr.w   -(sp)
560:   moveq.l #2,d0
561:   add.w   d7,d0
562:   move.w  d0,-(sp)
563:   move.l  #$ffffff,-(sp)
564:   pea     (a6)
565:   move.w  #18,-(sp)
566:   trap    #$e        ; Bootsektor initialisieren
567:   lea     14(sp),sp
568:   ;
569:   move.w  #800,d0    ; Anzahl Sektoren auf Disk
570:   addq.l  #1,d7
571:   mulu    d7,d0   ; wenn 2-seitig, dann x2-> 1600 Skt.
572:   move.b  d0,$13(a6) ; Anzahlen Sektoren im 8086-Format
573:   lsr.w   #8,d0      ; abspeichern (LOW-HIGH)
574:   move.b  d0,$14(a6)
575:   move.b  #10,$18(a6) ; Anzahl Sektoren/Track
576:   moveq.l #1,d0       ; Puffer n.Sektor 1/Track 0sichern
577:   bsr     writesec
578:   bsr     clrpuf           ; Puffer löschen
579:   move.l  #$f7ffff00,(a6) ; FAT initialisieren
580:   moveq.l #7,d0            ; Puffer nach Sektor 7 und 2
```

```
581:   bsr    writesec
582:   moveq.l #2,d0
583:   bra    writesec
584: formend:
585:   rts
586:
587: ; schreibt Sektor D0 auf Disk (Track 0/Seite 1)
588: writesec:
589:   move.w #1,-(sp)    ; nur einen Sektor schreiben
590:   clr.w  -(sp)
591:   clr.w  -(sp)
592:   move.w d0,-(sp)    ; Sektornummer
593:   clr.w  -(sp)
594:   clr.l  -(sp)
595:   pea    (a6)
596:   move.w #9,-(sp)
597:   trap   #$e         ; Sektor schreiben
598:   lea    20(sp),sp
599:   rts
600:
601: ; Puffer für Track- und Sektorendaten löschen
602: clrpuf:
603:   move.l a6,a0
604:   move.w #512/4,d0
605: clrpuf1:
606:   clr.l  (a0)+       ; 512 Bytes löschen
607:   dbf    d0,clrpuf1
608:   rts
609:
610: ; Zeigt den Grafen des Samples im Speicher an. Dieser
     wird in den Bildschirmzeilen 18-25 dargestellt.
611: ; Die Pixelzeilen sind auflösungsabhängig.
612: showgraf:
613:   lea    clrgrtxt(pc),a0
614:   bsr    print         ; Zeilen 18-25 löschen
615:   move.l lineavar,a6 ; Basis der Grafikvariablen
616:   lea    38(a6),a6 ; Basis der X1,Y1/X2,Y2-Koordinaten
617:   move.l memadr,a5  ; Startadresse in A5
618:   move.l #640,d3    ; Breite einer Pixelzeile
619:   move.l smpllen,d7 ; Länge des Samples
620:   cmp.l  d3,d7      ; wenn Sample kürzer als 640 Bytes
621:   blt    showend    ; dann abbrechen
622:   divu   d3,d7      ; ansonsten Offset berechnen aus
623:   moveq.l #0,d6     ; Formel OFFS = LEN/640
624:   move.w d6,(a6)    ; Startkoordinaten X1 = 0
625:   move.w #18*16+56,d0 ; Y1 = Mittl.Zeile des Bereichs
626:   tst.w  rez         ; Zeile 18-25
627:   bne.s  high1   ; da auflösungsabhängig, b.mittlerer
628:   lsr.w  #1,d0   ; Auflösung Pixelzeile einfach
629: high1:             ; halbieren
630:   move.w d0,2(a6)    ; Y1 speichern
631: drgraf:              ; Einsprung für Spaltenzähler
632:   moveq.l #0,d5
633:   move.b (a5),d5     ; Samplebyte auslesen
634:   lea    0(a5,d7),a5 ; m.Offset z.nächsten Samplebyte
635:   lsr.b  #2,d5       ; nur Bit 2-7 verwenden
636:   move.b d5,d4  ; Samplebyte liegt im Wertbereich 0-63
637:   lsr.b  #2,d4  ; wir haben aber 112 Pixelzeilen zur
638:   add.b  d4,d5  ; Verfügung, darum x1.75 nehmen
639:   add.b  d4,d5  ; (1 + 1/4 + 1/4 + 1/4)
640:   add.b  d4,d5
641:   add.w  #18*16,d5 ; Absolute Pixelzeile berechnen
642:   tst.w  rez
643:   bne.s  highrez   ; wenn mittlere Auflösung, dann
644:   lsr.w  #1,d5     ; einfach Höhe halbieren
645: highrez:
646:   move.w d6,4(a6)  ; X2-Koordinate
647:   move.w d5,6(a6)  ; gerade berechnete Y2-Koordinate
648:   .dc.w  $a003     ; DRAW LINE
649:   move.w d6,(a6) ; End  werden Startkoordinaten
650:   move.w d5,2(a6)
651:   addq.l #1,d6     ; X-Koordinate erhohen
652:   cmp.w  d3,d6     ; schon Endspalte 640 erreicht?
653:   blt    drgraf    ; nein, dann weitermachen
654: showend:
655:   rts
656:
657:
658: ; ————————————————————
659: ; Datenteil
660: ; ————————————————————
661: .data
662: inittext:
663:   .dc.b  $1b,"E",$1b,"f",0
664: menutext:
665:   .dc.b  $1b,"Y",32+16,32+0,$1b,"d",$1b,"H"
666:   .dc.b  " ",$1b,"p"," ST Sound Sampler - "
667: memtext:.dc.b  "0000 KB frei ",$1b,"q",13,10,13,10
668:   .dc.b  " Software by M.Backschat -
              Hardware by F.Weiss",13,10
669:   .dc.b  13,10,13,10
670:   .dc.b  " F1.....Hören",13,10
671:   .dc.b  " F2.....Aufnehmen",13,10
672:   .dc.b  " F3.....Sample laden",13,10
673:   .dc.b  " F4.....Sample sichern",13,10
674:   .dc.b  " F5.....Vorwärts spielen",13,10
675:   .dc.b  " F6.....Rückwärts spielen",13,10
676:   .dc.b  " F7.....Aufnahme/Hörfrq. einstellen",13,10
677:   .dc.b  " F8.....Abspielfrq. einstellen",13,10
678:   .dc.b  " F9.....Disk formatieren",13,10
679:   .dc.b  " F10....Ende!",13,10,0
680:
681: clrgrtxt:
682:   .dc.b  $1b,"Y",32+18,32+0,$1b,"J",0
683: get1text:
684:   .dc.b  $1b,"Y",32+16,32+0,$1b,"e",0
685: get2text:
686:   .dc.b  $1b,"f",0
687: msg1text:
688:   .dc.b  $1b,"Y",32+16,32+0,$1b,"p",0
689: msg2text:
690:   .dc.b  $1b,"q",0
691: clrmtext:
692:   .dc.b  $1b,"Y",32+16,32+0,$1b,"l",0
693: deltext:
694:   .dc.b  $1b,"D"," ",$1b,"D",0
695: quittext:
696:   .dc.b  " Programm verlassen (j/n)? ",0
697: heartext:
698:   .dc.b  " SPACE zum Abbrechen ",0
699: rec1text:
700:   .dc.b  " SPACE zum Aufnahmestart ",0
701: rec2text:
702:   .dc.b  " ESC zum frühzeitigen Abbruch ",0
703: setrtext:
704:   .dc.b  2
705:   .dc.b  " Neue Aufnahme/Hörfrq. in KHz (3-22): ",0
706: setptext:
707:   .dc.b  2
708:   .dc.b  " Neue Abspielfrq. in KHz (3-30): ",0
709: setetext:
710:   .dc.b  " Frq. war nicht im 3-22/30 KHz-Bereich! ",0
711: playtext:
712:   .dc.b  " SPACE zum Abbrechen (ansonsten
             Wiederholung)",0
713: savetext:
714:   .dc.b  40
715:   .dc.b  " Sample sichern unter: ",0
716: saveetxt:
717:   .dc.b  " Zu geringe Diskkapazität - Sample wird
             gekürzt! ",0
718: loadtext:
719:   .dc.b  40
720:   .dc.b  " Lade Sample mit Namen: ",0
721: loadetxt:
722:   .dc.b  " Kann Sample nicht finden! ",0
723: formtext:
724:   .dc.b  1
725:   .dc.b  " Anzahl Seiten (1,2,RETURN ist Abbruch): ",0
726: .even
727:
728: ; Kommandoadressen, die  über F1-F10 aufrufbar sind
729: cmdtab:
730:   .dc.l  listen, record, load , save , playforw
731:   .dc.l  playback, setrec, setplay , format , quit
732:
733: ; Lautsprecherdatentabelle
734: voldat:
735: .dc.l      $08000000,$09000000,$0a000000
736: .dc.l      $08000000,$09000000,$0a000200
737: .dc.l      $08000000,$09000000,$0a000300
738: .dc.l      $08000200,$09000200,$0a000200
739: .dc.l      $08000500,$09000000,$0a000000
740: .dc.l      $08000500,$09000200,$0a000000
741: .dc.l      $08000600,$09000100,$0a000000
742: .dc.l      $08000600,$09000200,$0a000100
743: .dc.l      $08000700,$09000100,$0a000000
744: .dc.l      $08000700,$09000200,$0a000000
745: .dc.l      $08000700,$09000300,$0a000100
746: .dc.l      $08000800,$09000000,$0a000000
747: .dc.l      $08000800,$09000200,$0a000000
748: .dc.l      $08000800,$09000300,$0a000100
```

```
749:  .dc.l        $08000800,$09000400,$0a000100
750:  .dc.l        $08000900,$09000000,$0a000000
751:  .dc.l        $08000900,$09000200,$0a000000
752:  .dc.l        $08000900,$09000300,$0a000100
753:  .dc.l        $08000900,$09000400,$0a000100
754:  .dc.l        $08000900,$09000500,$0a000000
755:  .dc.l        $08000900,$09000500,$0a000200
756:  .dc.l        $08000900,$09000600,$0a000000
757:  .dc.l        $08000900,$09000600,$0a000200
758:  .dc.l        $08000a00,$09000200,$0a000000
759:  .dc.l        $08000a00,$09000200,$0a000200
760:  .dc.l        $08000a00,$09000400,$0a000100
761:  .dc.l        $08000a00,$09000500,$0a000000
762:  .dc.l        $08000a00,$09000500,$0a000200
763:  .dc.l        $08000a00,$09000600,$0a000100
764:  .dc.l        $08000a00,$09000600,$0a000300
765:  .dc.l        $08000b00,$09000100,$0a000000
766:  .dc.l        $08000b00,$09000200,$0a000100
767:  .dc.l        $08000b00,$09000300,$0a000100
768:  .dc.l        $08000b00,$09000400,$0a000100
769:  .dc.l        $08000b00,$09000500,$0a000100
770:  .dc.l        $08000b00,$09000600,$0a000000
771:  .dc.l        $08000b00,$09000600,$0a000200
772:  .dc.l        $08000b00,$09000700,$0a000000
773:  .dc.l        $08000b00,$09000700,$0a000100
774:  .dc.l        $08000b00,$09000700,$0a000300
775:  .dc.l        $08000b00,$09000700,$0a000400
776:  .dc.l        $08000b00,$09000800,$0a000100
777:  .dc.l        $08000b00,$09000800,$0a000300
778:  .dc.l        $08000b00,$09000800,$0a000400
779:  .dc.l        $08000b00,$09000800,$0a000500
780:  .dc.l        $08000b00,$09000800,$0a000500
781:  .dc.l        $08000c00,$09000200,$0a000000
782:  .dc.l        $08000c00,$09000200,$0a000200
783:  .dc.l        $08000c00,$09000400,$0a000100
784:  .dc.l        $08000c00,$09000500,$0a000000
785:  .dc.l        $08000c00,$09000500,$0a000300
786:  .dc.l        $08000c00,$09000600,$0a000000
787:  .dc.l        $08000c00,$09000600,$0a000200
788:  .dc.l        $08000c00,$09000700,$0a000000
789:  .dc.l        $08000c00,$09000700,$0a000300
790:  .dc.l        $08000c00,$09000700,$0a000400
791:  .dc.l        $08000c00,$09000800,$0a000000
792:  .dc.l        $08000c00,$09000800,$0a000300
793:  .dc.l        $08000c00,$09000800,$0a000400
794:  .dc.l        $08000c00,$09000800,$0a000500
795:  .dc.l        $08000c00,$09000900,$0a000000
796:  .dc.l        $08000c00,$09000900,$0a000300
797:  .dc.l        $08000c00,$09000900,$0a000400
798:  .dc.l        $08000c00,$09000900,$0a000500
799:
800:  siditab:
801:    .dc.b   0,$ff,1,$ff,2,$ff,3,$ff,4,$ff,5,$ff,6,0,7,$3f
802:    .dc.b   8,0,9,0,10,0, $ff,0
803:
804:  recfrq:
805:    .dc.w   20000
806:  playfrq:
807:    .dc.w   20000
808:  smpllen:
809:    .dc.l   1
810:
811:
812:  ; ────────────────────────
813:  ; Datenteil uninitialisiert
814:  ; ────────────────────────
815:  .bss
816:  rez:
817:    .ds.w   1
818:  lineavar:
819:    .ds.l   1
820:  oldsp:
821:    .ds.l   1
822:  inptext:
823:    .ds.b   80
824:  frqptr:
825:    .ds.l   1
826:  memadr:
827:    .ds.l   1
828:  freemem:
829:    .ds.l   1
830:  startadr:
831:    .ds.l   1
832:  endadr:
833:    .ds.l   1
834:  diskinfo:
835:    .ds.l   4
836:  formtpuf:
837:    .ds.b   9*1024
```

Das Assemblerlisting des Sound-Samplers

ENDE

ERWEITERUNGEN FÜR TOY PROLOG

Durch den Public-Domain-Service erhält jeder Atari-ST-Besitzer die Möglichkeit, die Programmiersprache Prolog zu nutzen.

Auf der Diskette PD 11 ist die Implementation TOY PROLOG ST zu haben. Nun sind verschiedene Interpreter anderer Rechner mit einer größeren Zahl vordefinierter Prädikate ausgestattet. Dadurch wird die Portabilität von Prolog-Programmen eingeschränkt. Ein großer Teil des TOY-Prolog ist aber bereits in Prolog selbst geschrieben. Wir können daher genauso einige weitere Prädikate in Prolog schreiben und dem Interpreter hinzufügen. Die Programmierung solcher Prädikate soll in diesem Artikel besprochen werden.

Programmiersprachen wie Pascal, BASIC oder Fortran bestehen aus einer Folge von Anweisungen, die der Computer erfüllen soll. Da heißt es z.B.' Drucke "Hallo"' oder' Gehe nach 200'. Prolog hingegen baut auf dem Begriff des Prädikats auf. Ein solches kann man sich am ehesten als Frage vorstellen. Nehmen wir zur Veranschaulichung das Prädikat sum. Rufen wir es auf als 'sum(2,3,5)', stellen wir die Frage:' Ist 2+3=5?' Da das richtig ist, erhalten wir die Antwort' yes'. Ganz analog wird dagegen die Frage 'sum(1,3,5)' mit'no' beantwortet.

Wollen wir sum zur Berechnung einer Summe heranziehen, müssen wir eine Variable benutzen, 'sum(1,3,X)'. Damit fragen wir: 'Für welches X gilt 1+3=X?' Der Interpreter sucht dann eine Lösung und findet X=4. Genauso können wir fragen: 'Für welches X gilt 2+X=7?' Dazu müssen wir 'sum(2,X,7)' eingeben und bekommen die Antwort 'X=5.'

Variablen beginnen übrigens mit einem großen Buchstaben, Prädikate mit einem kleinen. Zur Syntax ist weiter noch zu sagen, daß jede Anfrage mit einem Punkt abgeschlossen wird. Ist eine Lösung für eine Frage gefunden worden, wird diese ausgegeben, und der Interpreter wartet, was er nun tun soll. Soll er eine weitere Lösung des Problems suchen, dann muß ein ";" eingegeben werden, ansonsten ein beliebiges anderes Zeichen, beispielsweise ein Punkt.

Sehen wir uns nun einen konkreten Anwendungsfall an. Hierzu nehmen wir an,daß eine Anzahl von Prädikaten zum Thema Fußball in der Datensammlung vorhanden ist. Wir wollen nun wissen, welche deutsche Fußballmannschaft nach 1945 viermal Pokalsieger und einmal deutscher Meister war. Dazu stellen wir die Anfrage 'pokalsieger (X,4), meister (X,1)'. Diese wird beantwortet mit 'X=eintracht_frankfurt'.

Die Vorgehensweise eines Prolog-Interpreters bei der Abarbeitung einer solchen Programmzeile ist relativ schnell erklärt: Zunächst wird versucht, die erste Frage zu beantworten, also hier: Welche Mannschaft wurde viermal Pokalsieger? Da kann es durchaus mehrere Lösungen geben, z.B. werde als erste 'X=fc_köln' gefunden. (Pokalsieger 1968, 1977, 1987, 1983.) Damit wird diese Lösung an die Variable X gebunden und dann versucht, das nächste Prädikat mit dieser Lösung zu erfüllen. Es wird also die Anfrage 'meister(fc_köln)' gestellt. War der FC Köln (genau) einmal deutscher Meister? Antwort: Nein, er war dreimal Meister (1962, 1964, 1978).

Die Anfrage kann also nicht erfüllt werden. Somit geht der Interpreter zurück zur vorherigen Anfrage und sucht eine neue Lösung für diese. (Dies nennt man Backtracking.) Als nächstes findet er' X=eintracht_frankfurt' (Pokalsieger 1974, 1975, 1981, 1988). Damit kann er wieder nach rechts gehen und die Anfrage 'meister(eintracht_frankfurt,1)' stellen. Diesmal wird ein' yes' die Antwort sein, Eintracht Frankfurt war nur 1959 deutscher Meister. Weil kein weiteres Prädikat zu erfüllen ist, kann die Lösung ausgegeben werden. Soll noch eine weitere Lösung gesucht werden, geht der Interpreter wieder zum vorherigen Prädikat zurück. Dort findet er aber keine mehr, daher wird' no' geantwortet.

Obwohl sich das alles recht plausibel anhört, erfordert eigenes Programmieren in Prolog doch ein totales Umdenken und vor allem viel Übung. Oft hat es sich gezeigt, daß Beispiele sehr hilfreich beim Erlernen einer Programmiersprache sind. Wer Prolog allerdings von Grund auf lernen will, der sollte sich ein Lehrbuch zulegen, Literaturhinweise finden sich am Ende des Artikels. Denn dies soll kein Einführungsartikel in Prolog sein. Wer Prolog schon sicher beherrscht, kann hier neue Anregungen finden. Wer noch nicht so sicher im Umgang mit der Sprache ist, kann etwas dazulernen. Und wer schließlich Prolog noch gar nicht kennt, wird immerhin sehen, was man mit diesem System alles machen kann.

Wie bereits erwähnt, beziehen wir uns hier vollkommen auf TOY PROLOG ST. Wir werden dem Interpreter einige Prädikate hinzufügen, die TOY-Prolog an andere Implementationen annähern, sowie einige, die das Arbeiten mit Toy-Prolog erleichtern.

Neue Listenprädikate

Eine wichtige Datenstruktur in Prolog ist die Liste, die man sich am einfachsten als Aufzählung von Daten vorstellen kann. Zur Darstellung einer Liste gibt es verschiedene Notationen. Am bequemsten

benutzt man die eckigen Klammern, z.B. [a,s,d]. Einige Prädikate für Listen sind in Listing 1 angegeben.

first(L,X) ist wahr, falls X das erste Element der Liste L ist. Was wir hierbei vor uns haben, ist eine Regel. Der Operator :- ist zu interpretieren als "falls". Die gesamte Regel würde so lauten: first(L,X) ist wahr, falls L=[X|R] wahr ist. Was bedeutet [X|R]? Durch die eckigen Klammern wird wieder eine Liste gekennzeichnet. "|" ist der Kopf-Rest-Separator. D.h. die Variable X links davon kennzeichnet das erste Element, die Variable R rechts die Restliste. WÜrden wir z.B. die Anfrage [a,s,d,f]=[K|R] stellen, würde sie mit K=a, R=[s,d,f] beantwortet. Der Wert, den R durch die Anfrage erhält, ist für uns unwichtig. Deshalb taucht er auch nur auf der rechten Seite der Regel auf.

append(L,M,N) ist ein Prädikat, das bei den meisten Interpretern bereits vorhanden ist. Hier wird getestet, ob N die Liste ist, die durch Aneinanderhängen der Listen L und M entsteht. Natürlich können an jeder Stelle Variablen stehen.

append wird durch zwei Regeln rekursiv realisiert. Die erste Regel besagt, daß die leere Liste [] verkettet mit einer beliebigen Liste L wieder L ergibt. Bei der Lösungssuche wird zuerst versucht, diese Regel zu erfüllen. Dies wird im allgemeinen nicht gelingen, da der Benutzer meistens nichtleere Listen angeben wird. Da aber durch die Rekursion irgendwann dieser Fall doch auftritt, benötigen wir diese Regel als Randbedingung. Das Ausrufezeichen heißt in Prolog "Cut". Es bewirkt hier, daß nur eine Lösung gefunden werden soll. Es wären durchaus mehrere möglich, wenn append mit Variablen in den ersten beiden Argumenten aufgerufen wird. Aber durch den Cut wird das Auffinden weiterer Lösungen unterbunden.

Die zweite Regel lautet nun: append([K|R1],L,[K|R2]) ist wahr, falls append(R1,L,R2) wahr ist. D.h. die Anfrage wird auf den einfacheren Fall zurückgeführt, bei dem die ersten Elemente der jeweiligen Listen abgetrennt sind, was wieder mit dem Kopf-Rest-Separator gemacht wird. Dies geht jedoch nur dann gut, wenn beide Kopf-Elemente tatsächlich identisch sind, denn sie werden durch dieselbe Variable K repräsentiert. Andernfalls schlägt die Anfrage fehl.

Beim Prädikat last(L,X), das uns das letzte Element einer Liste liefert, verwenden wir append: last(L,X) ist wahr, falls L durch Aneinanderhängen einer beliebigen Liste und [X] entsteht. Auf ähnliche Art hätten wir auch first definieren können. Aber die Verwendung von append hätte Geschwindigkeit bei der Ausführung gekostet. Bei last hingegen kommen wir an einer Rekursion nicht vorbei, da eine Liste intern baumartig dargestellt ist und das letzte Element das am weitesten von der Wurzel stehende Element des Baums ist.

Das I-te Element X der Liste L erhalten wir durch item(I,L,X). Die Randbedingung sagt diesmal aus, daß X das erste Element der Liste ist, falls X der Kopf der Liste ist, so formuliert eine triviale Aussage. Der Cut sorgt wieder dafür, daß nur eine Lösung gesucht wird. Der Rest geschieht wie bei append rekursiv. Um die Listenelemente zu zählen, verwenden wir hier eine Hilfsvariable J. Diese wird um 1 kleiner als I. Anstelle von = müssen wir hier den Operator is verwenden, da is arithmetische Ausdrücke auswertet, = aber nicht. Die Rekursion reduziert die Anfrage jeweils auf den einfacheren Fall einer um 1 kürzeren Liste. Dies geht so lange, bis X als erstes Element einer Liste dasteht oder die Liste durch die Rekursion leer geworden ist.

Listing 2 zeigt, auf welche verschiedenen Arten die neu definierten Prädikate aufgerufen werden können. Die Zeichen ?- werden vom Interpreter vorgegeben und erinnern den Benutzer daran, daß er nun eine Anfrage stellen kann. Das letzte Beispiel zeigt den Aufruf der neu definierten Prädikate für Ein/Ausgabe, die nun besprochen werden sollen.

```
1:  append([],L,L) :- !.
2:  append([K|R1],L,[K|R2]) :- append(R1,L,R2).
3:
4:  first(L,X) :- L=[X|R].
5:
6:  last(L,X) :- append(H,[X],L).
7:
8:  item(1,[X|R],X) :- !.
9:  item(I,[K|R],X):- J is I - 1, item(J,R,X).
```

Listing 1: Prädikate für Listen

```
1:  ?-append([a, s], [d], [a, s, d]).
2:  yes
3:  ?-append([a, s], [d], [a, s, f]).
4:  no
5:  ?-append([a, s], [d], X).
6:
7:  X = [a, s, d] .
8:  ?-append([a, s], X, [a, s, d, f]).
9:
10: X = [d, f] .
11: ?-append(X, Y, [a, s, d]).
12:
13: X = []
14: Y = [a, s, d] .
15: ?-first([a, s, d], [a]).
16: no
17: ?-first([a, s, d], a).
18: yes
19: ?-first([a, s, d], X).
20:
21: X = a .
22: ?-last(X, a).
23:
24: X = [a] .
25: ?-last([a, s, d], X).
26:
27: X = d .
28: ?-item(3, [a, s, d, f], X).
29:
30: X = d .
31: ?-item(8, [a, s, d], X).
32: no
33: ?-item(8, X, a).
34:
35: X = [X0, X1, X2, X3, X4, X5, X6, a | X7] .
36: ?-write(a) , tab(20) , writeln(b).
37: a                   b
38: yes
```

Listing 2 : Beispielaufrufe der Listenprädikate

geht weiter...

Prädikate für Ein/Ausgabe

Die genauen Definitionen sind in Listing 3 angegeben. In Anlehnung an Pascal definieren wir zunächst das Prädikat writeln(X), welches nach Ausgabe von X einen Zeilenvorschub durchführt. In Toy-Prolog wird ein Zeilenvorschub durch nl gekennzeichnet. Das Prädikat tab(X) gibt X Leerzeichen aus. Die Definition erfolgt wieder rekursiv. Als Randbedingung geben wir tab(0) an. Die dritte, rekursive Regel gibt jeweils ein Leerzeichen mittels wch aus und bezieht sich dann auf den nächst niedrigeren Fall. Dies geht so lange, bis tab(0) aufgerufen wird. Bei Aufruf mit einem negativen Wert würde die Rekursion so bis ins Unendliche gehen. Deshalb müssen wir da einen Riegel vorschieben und diesen Fall ausschließen.

In der zweiten Regel wird zunächst getestet, ob X<0 ist. Ist das der Fall, gelangt der Interpreter zu einer sogenannten Cut/Fail-Kombination. Diese bewirkt, daß die Anfrage fehlschlägt, aber auch keine weiteren Lösungen gesucht werden.

```
1:  tab(0).
2:  tab(X) :- X<0, !, fail.
3:  tab(X) :- Y is X - 1, wch(' '), tab(Y).
4:
5:  writeln(X) :- write(X), nl.
```

Listing 3 : Prädikate für Ein/Ausgabe

Listing von Prädikaten

In Prolog ist es möglich, mit dem Prädikat listing den Text sämtlicher Prädikate aufzulisten. Ein unglücklicher Umstand bei Toy-Prolog ist es, daß hierdurch auch die Systemprädikate betroffen sind, die ja auch in Prolog geschrieben sind. Anstelle eines kurzen Listings der wenigen selbstdefinierten Prädikate, an denen man gerade arbeitet, erhält man so fast den gesamten Inhalt der Systemdatei ausgegeben. Was wir nun selbst programmieren wollen, ist ein Prädikat, das nur die von uns definierten Prädikate listet.

Dazu müssen wir einen kleinen Eingriff in das System vornehmen. Wir nehmen einmal an, wir lesen alle unsere Prädikate jeweils mit consult ein. Dann definieren wir einfach ein neues Prädikat load, welches genau dasselbe wie consult erledigt, aber zusätzlich jedes eingelesene Prädikat markiert. Das Prädikat list wird dann nur diejenigen Prädikate auflisten, die mittels load markiert wurden.

Glücklicherweise müssen wir uns die Programmierung von load nicht von Grund auf neu überlegen, denn mit listing können wir ja den Text von consult erhalten. In Listing 4 sehen wir das Ergebnis. Die Prädikate see, seeing und seen öffnen und schließen Dateien, echo und noecho sorgen für die Wiedergabe auf dem Bildschirm. Von Interesse ist für uns nur getprog. Er stellt die Schleife dar, in der die Prädikate eingelesen und definiert werden.

Wie wir das nun verwenden müssen, können wir in Listing 5 sehen. load schreiben wir wie consult, ersetzen aber readprog durch readprogram. Dieses wird wie readprog definiert, jedoch getprog wird nun getprogram genannt. getprogram schließlich wird wie getprog definiert, nur schieben wir make_listable(X) ein. Dies sorgt dafür, daß das Prädikat P, das durch die Regel X definiert ist, markiert wird. Die Markierung geschieht durch Hinzufügen der Klausel listable(P) zur Datenbank. (Eine Klausel ist eine Regel ohne Bedingung.) Weil jede Datei durch end abgeschlossen werden muß, wird somit auch jedesmal ein listable(end) der Datenbank hinzugefügt. Dies ist aber nicht erwünscht, durch retract (listable(end)) wird es wieder entfernt.

make_listable(X) holt sich den Namen P des Prädikats, das durch X definiert wird, und prüft, ob listable(P) bereits vorhanden ist. Wenn nein, wird es hinzugefügt.

Auf das Prädikat pred(X,K) soll nicht so ausführlich eingegangen werden. Es prüft, ob K der Name des Prädikats ist, das durch die Regel X definiert wird. Eine Fallunterscheidung muß vorgenommen werden. Denn einmal kann X eine Regel mit dem Operator :- sein, zum anderen eine grammatische Regel mit dem Operator —> . Außerdem ist es möglich, daß X eine Klausel ist. Das Prädikat list holt sich mit listable(X) jeweils ein markiertes Prädikat und gibt mittels listing(X) dessen Programmtext aus. Das fail am Ende sorgt dafür, daß die Anfrage fehlschlägt und durch Backtracking nacheinander alle Lösungen von listable(X) gefunden werden.

Fassen wir also zusammen: Laden wir eine Datei mit load, werden alle Prädikate daraus markiert. list gibt eine Auflistung nur der markierten Prädikate. Wollen wir Eingaben im Direktmodus machen, so können wir eine Markierung auch erreichen. Wir müssen nur load(user) eingeben. Außerdem kann ein Prädikat P auch nachträglich markiert werden. Nötig ist ein Hinzufügen von listable(P) zur Datenbank.

Aus Gründen der Einheitlichkeit definieren wir zusätzlich list(P), welches nur den Text des Prädikats P listet. listable gibt eine Liste aller markierten Prädikate. Da die Lösungen von listable(X), das als erstes aufgerufen wird, explizit in der Datenbank stehen, werden sie vom Interpreter leicht gefunden.

oblist gibt die Namen aller in Prolog definierten Prädikate aus, also auch die aus der Systemdatei. Erzeugt werden sie als Lösungen der Anfrage proc(X), die restlichen Prädikate aus dieser Regel dienen nur der optischen Aufbereitung, so etwa Abtrennen der Argumente etc. oblist arbeitet ebenso wie listable und list mit der Schleifenbildung durch fail.

Implementierung des Break

Die Anweisung break unterbricht den Programmlauf und führt den Benutzer wieder auf die oberste Interpreterebene. Zur Kennzeichnung wird allerdings in eckigen Klammern das Break-Level angezeigt, denn Verschachtelungen sind

```
1:  ?-listing(consult).
2:  consult(X1) :- seeing(X2) , readprog(X1) , see(X2).
3:
4:  yes
5:  ?-listing(readprog).
6:  readprog(user) :- ! , getprog.
7:  readprog(X1) :- see(X1) , echo , getprog , noecho , seen.
8:
9:  yes
10: ?-listing(getprog).
11: getprog :- repeat , read(X1) , assimilate(X1) ,X1 = end , !.
12:
13: yes
```

Listing 4 : Definition von consult

MUSIKER:
ABENTEUER:
SCHACHMATT!
Juli/August '87 DM 6,– 8

NUR AUFGEBLASEN?
Der neue Mega ST
K-MAX:
APL 68000:
Eine Zeichensprache
OMIKRON BASIC COMPILER!
NEU NEU NEU GRAFIK KURS 1. TEIL
September '87 DM 6,– 9

Die Fachzeitschrift für den ATARI-ST Anwender
PUZZLE
ALICE IM WUNDERLAND
INS NETZ GEGANGEN
TRUE BASIC
Oktober '87 DM 6,– 10

Die Fachzeitschrift für den ATARI-ST Anwender
November '87 DM 7,– 11
ÜBERSICHT
5 Datenbanken im Vergleich
STEVE
Alles in Einem
FLOATING POINT PROCESSOR
68881-Rechner

Dezember '87 DM 7,– 12

Die Fachzeitschrift für den ATARI-ST Anwender.
Januar '88 DM 7,– 1
KLINGENDE CHIPS
Sound-Digitalisierer
FARBVIELFALT
Mit dem Omega-Grafik-System
SCHWARZ AUF WEISS
Der neue NEC-Multisync GS
KLEINE HELFER
Hard Disck Utilities auf dem Prüfstand
NEWS AUS USA
Comdex '87

2
DER BIT BRENNER
Für den Selfmade Man
CHESS-BASE
ZEICHNEN AUF EINEM TABLETT
FORTRAN-77
im neuen Gewand

3
ALS DIE BILDER LAUFEN LERNTEN
Video 8 AF

April '88 DM 7,– 4
VT52-EMULATOR
NEUE SERIE

5
CeBIT'88
Was gibt's Neues bei ATARI ?
GFA-BASIC3.0
Noch schneller noch besser?
SM124 MULTISYNC
Und es geht doch !
MUSIKMESSE FRANKFURT
Alle Neuigkeiten für den ST
ATARI NEWS

6
TRANSPUTER
STAR-WRITER
Computerbild auf Video
Aufrüstung der ATARI-Festplatte auf 40 MB

Die Fachzeitschrift für den ATARI-ST Anwender
Juli '88 DM 7,– 7
PUZZLESPRACHE
CAMPUS DRAFT
DISKMASTER
MODULA 2



Einzelheft- u. Monatsdisketten Bestellung

Heim Verlag
Heidelberger Landstr. 194
6100 Darmstadt-Eberstadt
Telefon 06151-56057



Einzelheft- u. Monatsdisketten **Bestellung**

ST-Computer können Sie direkt beim HEIM-VERLAG zum Einzelheft-Preis von DM 6,– (1987) DM 7,– (1988) nachbestellen. Bearbeitung nur gegen beigefügten Scheck über den entsprechenden Betrag (keine Überweisung).

Jan.	Febr.	März	April	Mai	Juni	Juli/Aug.	Aug./Sept.	Okt.	Nov.	Dez.	
											1987 – DM
											1988 – DM

+ Gebühr für Porto und Verpackung – DM

(1 Heft DM 2,–, ab 2 Hefte DM 5,–)

☐ Scheck in Höhe zus. DM liegt bei

Disketten Service

Alle Programme, die in ST-Computer veröffentlicht wurden, sind auf Disketten erhältlich. Die Disketten enthalten die Programme von jeweils 2 ST-Computer-Ausgaben. Bestellen Sie durch ankreuzen die gewünschten Disketten

Preis je Diskette 28,– DM	Jan./Febr.	März/Apr.	Mai/Juni	Juli/Aug.	Sept./Okt.	Nov./Dez.	Jan./Febr.	März/Apr.	Mai/Juni
	87	87	87	87	87	87	88	88	88
	Juli/Aug.	Sept./Okt.	Nov./Dez.	Jan./Febr.	März/Apr.	Mai/Juni	Juli/Aug.	Sept./Okt.	Nov./Dez.
	88	88	88	89	89	89	89	89	89

Lieferung: gegen beigefügten Scheck zuzügl. 5,– DM Versandkosten, unabhängig der bestellten Menge.

möglich. Nun können beliebige Anfragen gestellt werden. Durch Eingabe von cont gelangt man wieder ein Level tiefer. Das dort begonnene Programm wird nun fortgeführt.

Um dies nachträglich in Prolog zu programmieren, müssen wir uns dieses Mal die Interpreterschleife ansehen. Der Dokumentation zu Toy-Prolog entnehmen wir, daß der sogenannte äußere Interpreter durch das Prädikat ear gestartet wird. Mit listing verschaffen wir uns wieder den Programmtext (siehe Listing 6).

ear ruft loop auf. In der Schleife loop wird die Benutzereingabe gelesen und ausgeführt. Ein paar Worte müssen zu dem Aufruf von loop gemacht werden. loop ist eine Endlosschleife. Dafür sorgt die repeat/fail-Konstruktion. Riefe man dieses Prädikat einfach durch Angabe des Namens auf, würde es nie zum Ende kommen. Das ist aber nicht erwünscht. Durch den Aufruf tag(loop) erhalten wir die Möglichkeit, später von außen auf die Erfüllung des Prädikats loop Einfluß zu nehmen. Denn ein Aufruf von tagfail(loop) bewirkt, daß loop sofort fehlschlägt, was damit auch für ear gilt. Der Interpreter versucht dann, ear mittels der zweiten Regel zu erfüllen. Dort wird durch halt der Interpreter abgebrochen.

Die Idee zur Programmierung des Break ist nun die folgende: Das Prädikat break wird analog zu ear programmiert. Die ausgegebenen Meldungen werden jedoch anders aussehen. ear ruft statt loop das Prädikat break_loop auf. Dieses sieht fast genauso wie loop aus, nur wird das Break-Level in eckigen Klammern mit ausgegeben. cont entspricht stop.

Die Realisierung finden wir in Listing 7. Zur Kennzeichnung des Break-Levels existiert jeweils ein Prädikat break_level(X). Durch Aufruf von diesem wird X auf den entsprechenden Wert gesetzt. Geht man in ein anderes Break-Level, muß dieses Prädikat gelöscht und durch das mit dem neuen Wert ersetzt werden. So ist zu Beginn das Prädikat break_level(0) vorhanden. Bei Aufruf von break wird es gelöscht und durch break_level(1) ersetzt. Eingabe von cont macht dies wieder rückgängig.

Listing 8 zeigt uns die Wirkungsweise von break. Nachdem das Atom 'eins' ausgegeben wurde, gelangen wir auf Break-Level 1. Dort können wir uns 'zwei' ausgeben lassen. Nach Eingabe von cont gelangen wir zurück. Dort muß die angefangene Programmzeile noch beendet werden, d.h. 'drei' wird ausgegeben.

```
 1:  readprogram(user) :- !, getprogram.
 2:  readprogram(F) :- see(F), echo, getprogram, noecho, seen.
 3:
 4:  pred(X,K) :- X=..L, L=[':-'|[H|R]], H=..[K|Q], !.
 5:  pred(X,K) :- X=..L, L=['->'|[H|R]], H=..[K|Q], !.
 6:  pred(X,K) :- X=..L, L=[K|R].
 7:
 8:  make_listable(X) :- pred(X,K), not(listable(K)),
 9:                      assertz(listable(K)).
10:
11:  getprogram :- repeat, read(X), assimilate(X),
12:                make_listable(X), X=end,
13:                retract(listable(end)), !.
14:
15:  load(F) :- seeing(G), readprogram(F), see(G).
16:
17:  list(X) :- listing(X).
18:  list :- listable(X), listing(X), fail.
19:
20:  listable :- listable(X), write(X), write('  '), fail.
21:
22:  oblist :- proc(X), X=..L, L=[K|R], write(K), write('  '),
                                                        fail.
```

Listing 5 : Zum Listen von Prädikaten

Einen echten Anwendungsfall sehen wir in Listing 9. Angenommen, wir haben das Prädikat tier für eine Anzahl von Daten in der Datenbank. tier(elefant) beispielsweise soll heißen: Der Elefant ist ein Tier. Die Aufgabe soll darin bestehen, für jedes Tier, das vorhanden ist, ein Prädikat ordnung einzugeben, das die biologische Ordnung des jeweiligen Tieres angibt. Dazu können wir den Break benutzen.

Zur Illustration geben wir zunächst mit assert drei Daten ein. Bei der folgenden Anfrage findet tier(X) jeweils eins dieser Tiere, write(X) gibt es aus. Der dann folgende Break bringt uns auf die Interpreterebene, in der wir Eingaben machen können. Wir fügen dann das entsprechende Prädikat ein. Zuerst wird fledermaus gefunden. Wir wissen, die Fledermaus gehört zur Ordnung der Flattertiere und geben dies ein. Nach cont wird die Anfrage fortgesetzt, wir erhalten das nächste Tier. Das fail bildet wieder eine Schleife.

Einbau der neuen Prädikate

Das Toy-Prolog-System erlaubt es nun, die neu definierten Prädikate der Systemdatei hinzuzufügen, so daß sie nach Starten des Systems direkt zur Verfügung stehen. Wie das gemacht wird, wollen wir jetzt noch besprechen.

Zunächst schreiben wir mit einem Editor alle Prädikate, die wir in unserem System neu haben wollen, in eine einzige Datei. Wir nennen sie jetzt einmal ERWEIT. Die letzte Zeile bildet die Anweisung end. , der Punkt darf nicht fehlen. Jetzt starten wir den Prolog-Interpreter und rufen das Prädikat translate auf. Hiermit übersetzen wir den von uns geschriebenen Code in die Syntax des inneren Interpreters. Der Aufruf lautet translate(erweit,erw). Der gebildete Code wird in der Datei ERW abgelegt.

Mit stop beenden wir Prolog und starten wieder den Editor. Hier laden wir zunächst die Datei ERW und löschen deren letzte drei Zeilen. Diese bestehen aus einem Doppelpunkt, dem Wort seen und der Zeichenkombination []#. Nun laden wir die Datei SYSFILE.TOY und gehen

```
 1:  ?-listing(ear).
 2:  ear :- nl , display('TOY Prolog listening:') , nl ,tag(loop).
 3:  ear :- grf_mode , halt('TOY Prolog — end of session.').
 4:
 5:  yes
 6:  ?-listing(loop).
 7:  loop :- repeat , display(?-) , read(X1, X2) , exec(X1,X2) , fail.
 8:
 9:  yes
10:  ?-listing(stop).
11:  stop :- tagfail(loop).
12:
13:  yes
```

Listing 6 : Definition von ear

mit dem Cursor an das Ende des Textes. Dort finden wir nach einem Doppelpunkt einige Anweisungen, die direkt ausgeführt werden sowie die Definition des letzten Prädikats, unprotect. Dieses endet mit []. Zwischen [] und den Doppelpunkt wird die geänderte Datei ERW nun geladen.

Das war fast schon alles. Wie wir aber sehen, werden durch protect alle geladenen Prädikate geschützt. Das Prädikat break_level wird jedoch bei Aufruf von break verändert, was bei einem geschützten Prädikat nicht möglich ist. Deshalb müssen wir nur hierfür den Schutz rückgängig machen. Wie die letzten Zeilen von SYSFILE.TOY aussehen sollten, zeigt Listing 10.

Eckart Winkler

Literatur :

1.) Bratko : Prolog - Programmierung in der Künstlichen Intelligenz (Addison-Wesley)
2.) Giannesini : Prolog (Addison-Wesley)
3.) Grothaus : Turbo-Prolog (Chip Wissen)
4.) Hanus : Problemlösen mit Prolog (Teubner)
5.) Holland : Problemlösen mit Micro-Prolog (Teubner)
6.) Kleine Büning/Schmitgen : Prolog (Teubner)
7.) Mossakowski : Programmieren mit Turbo-Prolog (Markt & Technik)
8.) Schmitter/Flögel : Prolog-Anwendungen (Hofacker)
9.) Stender : Prolog-Handbuch (Hofacker)
10.) Dokumentation zu TOY PROLOG ST (als Datei auf PD 11)


```
 1:  ?-write(eins),break,writeln(drei).
 2:  eins
 3:  Break Level 1
 4:  [1]?-writeln(zwei).
 5:  zwei
 6:  yes
 7:  [1]?-cont.
 8:
 9:  End Break Level 1
10:  drei
```

Listing 8 : Funktionsweise von break

```
1:  :  'grf_mse_hide' .
2:     'txt_mode' .
3:     'protect' .
4:     'unprotect' ('break_level', 1) .
5:     'seen'  [] #
```

Listing 10 : Die letzten Zeilen der Systemdatei

ENDE

```
 1:  break :- break_level(X), Y is X + 1,
                                    retract(break_level(X)),
 2:            assert(break_level(Y)), nl,
 3:            write('Break Level '), write(Y), nl,
                                              tag(break_loop).
 4:  break :- break_level(X), Y is X - 1,
                                    retract(break_level(X)),
 5:            assert(break_level(Y)), nl,
 6:            write('End Break Level '), write(X), nl.
 7:
 8:  break_loop :- repeat, break_level(X),
 9:                write('['), write(X), write(']?-'),
10:                read(Y,Z), exec(Y,Z), fail.
11:
12:  cont :- tagfail(break_loop).
13:
14:  break_level(0).
```

Listing 7 : Implementierung des Break

```
 1:  ?-assert(tier(elefant)).
 2:  yes
 3:  ?-assert(tier(hamster)).
 4:  yes
 5:  ?-assert(tier(fledermaus)).
 6:  yes
 7:  ?-tier(X),write(X),break,fail.
 8:  fledermaus
 9:  Break Level 1
10:  [1]?-assert(ordnung(fledermaus,flattertier)).
11:  yes
12:  [1]?-cont.
13:
14:  End Break Level 1
15:  hamster
16:  Break Level 1
17:  [1]?-assert(ordnung(hamster,nagetier)).
18:  yes
19:  [1]?-cont.
20:
21:  End Break Level 1
22:  elefant
23:  Break Level 1
24:  [1]?-assert(ordnung(elefant,ruesseltier)).
25:  yes
26:  [1]?-cont.
27:
28:  End Break Level 1
29:  no
30:  ?-ordnung(X,Y).
31:
32:  X=elefant
33:  Y=ruesseltier ;
34:
35:  X=hamster
36:  Y=nagetier ;
37:
38:  X=fledermaus
39:  Y=nagetier ;
40:  no
```

Listing 9 : Anwendung von break

append(L1,L2,L)	- L ist die Verkettung der Listen L1 und L2
first(L,X)	- X ist das erste Element der Liste L
last(L,X)	- X ist das letzte Element der Liste L
item(I,L,X)	- X ist das I-te Element der Liste L
tab(N)	- Ausgabe von N Leerzeichen
writeln(X)	- Ausgabe von X mit Zeilenvorschub
pred(X,K)	- K ist der Name des Prädikats der Regel X
load(F)	- Laden des Files F und Markieren der Regeln
list(X)	- Listen der Regel X
list	- Listen aller markierten Regeln

Tabelle 1 : Die neu definierten Prädikate



TIMEWORKS

GST auf neuen Pfaden

GST ist wohl jedem Atari-Benutzer bekannt, das englische Softwarehaus ist schließlich für das Textprogramm 1st Word verantwortlich, unter dem wir alle zu leiden haben. Aber trotz seiner Macken gibt es immer noch keine vernünftige Alternative zu 1st Word plus. Viele haben sich gewundert, warum von GST schon so lange nichts mehr über Updates und Fehlerkorrekturen zu hören war. Gerüchte behaupteten, GST arbeite jetzt an einem DTP-Programm und das habe eben Priorität. Die Gerüchte hatten Recht - der Timeworks Desktoper Publisher wurde von GST für Timeworks Inc. entwickelt. So gut wie der Ventura Publisher oder PageMaker will der Timeworks Desktop Publisher sein. Wirklich?

Für 398.- DM erhalten Sie einen weißen Plastikschuber, in dem sich ein ebenfalls weißer Ringordner, der die Anleitung samt zwei Programmdisketten enthält, verbirgt.

Die Anforderungen, die Timeworks an die Gerätekonfiguration des künftigen Benutzues stellt, sind an sich gering; sogar auf einem 520 ST läuft der Publisher, allerdings muß wegen der Größe des Programmes eine Festplatte unbedingt empfohlen werden, schon allein deshalb, weil sehr viele Dateien geladen werden müssen, was doch immer einige Zeit beansprucht. Außerdem werden ständig Dateien von Diskette nachgeladen. Mit einem Disklaufwerk zu arbeiten, ist so gut wie unmöglich.

Anleitung und Programm liegen in deutscher Übersetzung vor, die im allgemeinen gut gelungen ist, und nur an manchen Stellen ist ihre Abkunft von einem englischen Original nicht zu übersehen. Die Qualität des Handbuchs ist, den Inhalt mitberücksichtigt, mittelprächtig, oft sind die Erklärungen, trotz großer Ausführlichkeit, nicht so klar und eindeutig wie es wünschenswert wäre.

Bitte Platz nehmen zum zweiten Frühstück

Bevor man sich den Publisher auch nur ansehen kann, muß das Programm an die vorhandenen Geräte (Computertyp, Drucker usw.) angepaßt werden. Dazu dient ein Installationsprogramm. Wenn man mit Diskettenlaufwerken arbeitet, sind zwei Disketten Ergebnis der Installation, eine davon dient zum Booten und Laden des Programms, während die andere Zeichensätze und Daten enthält. Die Boot-Diskette sollte auch nach dem Laden immer im Laufwerk A des Rechners bleiben, da, wie bereits erwähnt, immer wieder Programmteile nachgeladen werden müssen. Wenn Ihre Festplatte autoboot-fähig ist, können Sie das ganze Programm auch so auf der Festplatte installieren lassen, daß es komplett von dort gestartet werden kann.

Die Installation selbst auf Festplatte dauert, vorsichtig ausgedrückt, ewig, vielleicht auch etwas länger. Stellen Sie sich ein paar Brötchen bereit und die Morgenzeitung, Sie haben jetzt viel, viel Zeit, nur alle paar Sekunden müssen Disketten gewechselt werden, aber das sollte Sie nicht beim zweiten Frühstück stören.

Trotz der quälenden Länge der Prozedur ist die Anpassung aber sehr komfortabel und das Programm tut alles selbstständig (außer Disketten zu wechseln), so daß gerade Computerneulinge nicht mit Problemen wie "Erzeugen Sie jetzt auf Ihrer Startup-Disk einen neuen Ordner mit dem Namen Xy. In diesen Ordner kopieren Sie die und die Dateien und schreiben dann in die Info-Datei Yz den (vollständigen!) Pfadnamen, über den die Systemfonts zu finden sind" konfrontiert werden, sondern nur, bequem im Sessel zurückgelegt, in einer Dialogbox ein paar Fragen aus der Intimsphäre ihres Rechners zu beantworten brauchen. Bei dieser Prozedur entsteht automatisch auch gleich ein Backup der Original-Disketten, da die Dateien, aus denen das Programm besteht, bei der Installation auf frische (natürlich nicht mitgelieferte) Disketten kopiert werden.

Weniger bequem sind Änderungswün-

sche. Wenn Sie öfter einmal mit verschiedenen Druckern arbeiten wollen, müssen Sie schon zwei Installationen anlegen, was zumindest auf einer Festplatte, je nach Autoboot-Programm, nicht ganz unproblematisch sein wird. Ohne die gesamte Installation zu wiederholen, kann nämlich nicht einmal die Druckeranpassung geändert werden. In diesem Punkt kann man nur flehende Briefe an GST schreiben, mit der freundlichen Bitte um etwas mehr Flexibilität.

In Sachen Anpassung des Druckers ist es dagegen eine gute Idee, eine fertig produzierte DTP-Seite (auf Diskette) beizulegen, die ein Testmuster enthält, mit dem man sehr einfach den Drucker so einstellen kann, das der Ausdruck auch richtig auf der Seite zentriert ist.

INSTALL.APP

TIMEWORKS DESKTOP PUBLISHER - INSTALLATION

SYSTEM
Atari 520ST 0.5 Mb
Atari 1040ST 1 Mb
Atari MEGA ST2 2 Mb
Atari MEGA ST4 4 Mb

DISK
Ein 350 Kb Floppy
Zwei 350 Kb Floppy
Ein 700 Kb Floppy
Zwei 700 Kb Floppy
Harddisk C:
Harddisk D:
Harddisk E:
Harddisk F:

DRUCKER
Epson 9-Nadel 120x144 dpi
Epson 24-Nadel 180 dpi
NEC P6/P7 180 dpi
NEC P6/P7 360 dpi
HP LaserJet Plus 150 dpi
HP LaserJet 2 300 dpi
Atari SLM804 Laser 300 dpi
PostScript Laser 300 dpi

BILDSCHIRM
Mono 640x400
4-Farbig 640x200

INSTALL ABBRUCH

Bild 1: Das Installationsprogramm : Sie brauchen jetzt viel Zeit

Ran an den Speck

Der Timeworks Publisher ist ein rahmenorientiertes DTP-Programm. Das bedeutet, daß um jedes Layout-Element, ob es nun eine Textspalte oder eine Grafik ist, ein viereckiger Rahmen gezeichnet wird, in dem sich dann das weitere Geschehen (z.B. das Edieren von Texten) abspielt. Dabei können sich Rahmen auch überlappen. Im Prinzip geben die Rahmen die Aufteilung der Seite wieder. Im Grunde entspricht das der 'alten' Arbeitsweise mit Papier und Klebstoff. Früher ließ man sich den Text in Spaltenform ausdrucken, genauso die Überschriften und Grafiken, schnitt die Teile dann zurecht und klebte sie auf die Seite. Das Zer- und Zurechtschneiden der einzelnen Teile erledigt man mit einem rahmenorientierten DTP mit der Maus, genauso wie das Einkleben, entsprechend dem Plazieren der Teile auf der Seite. Mit der elektronischen Schere/Klebstoffflasche hat man natürlich ein paar Möglichkeiten mehr: Man kann Rahmen zum Beispiel einfach kopieren.

Im Timeworks Publisher sind Rahmen nicht immer ausschließlich 'Platzhalter' für irgendwelchen Inhalt, sie können auch einen gezeichneten Rahmen in verschiedenen Linienstärken haben oder den Rahmen mit Füllmustern ausfüllen, sozusagen als Rahmenhintergrund für den Inhalt. Nur dann werden die Rahmen gedruckt, normalerweise erscheinen sie aber nicht auf dem Papier, sondern dienen nur der Edierung. Man kann die Darstellung dieser Hilfsrahmen auch abschalten, um das Layout der Seiten 'so richtig' zu sehen. Der Timeworks Publisher arbeitet außerdem nach dem vielbeschworenen 'What You See Is What You Get'-Prinzip, übersetzt: Sie bekommen das, was Sie sehen. Gemeint ist natürlich nicht, daß Ihnen ab sofort jeder Wunsch, der beim Betrachten eines Gegenstandes in der belebten oder unbelebten Natur aufkommt, erfüllt wird, sondern, etwas kleinkarierter, daß Sie auf dem Papier, das aus Ihrem Drucker kommt, genau das sehen, was Sie vorher am Bildschirm zusammenmontiert haben. Wie genau das Layout am Bildschirm mit dem Papier-Ergebnis übereinstimmt, ist unter anderem ein Qualitätsmaßstab für ein DTP-Programm.

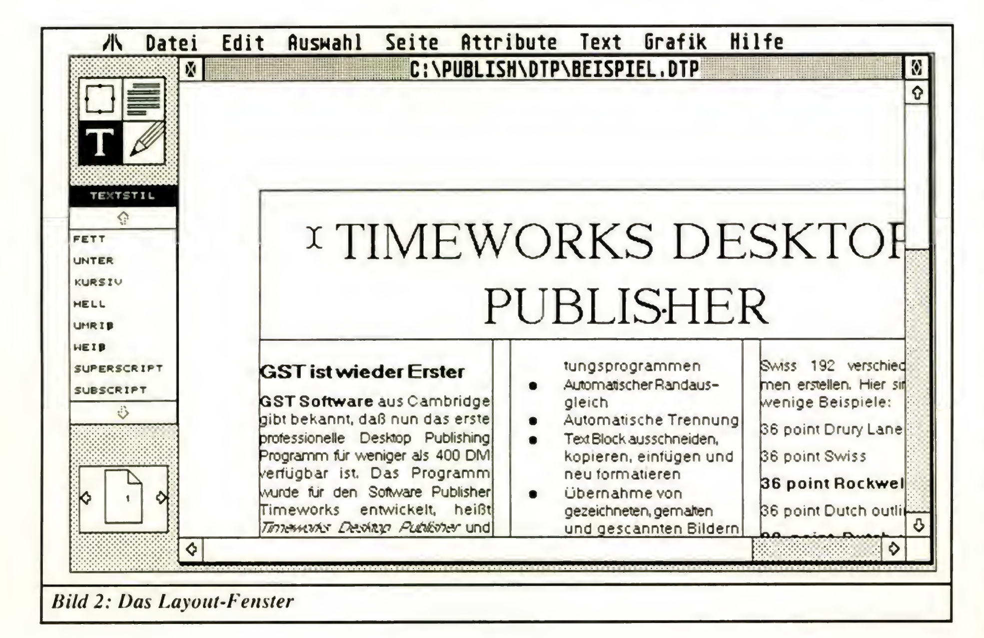


Bild 2: Das Layout-Fenster

Ein kleiner Editor

Sie können auf eine Seite, die Sie mit dem Programm bearbeiten, wild irgendwelche Rahmen zeichnen und Sie nach Belieben mit Text, Bildern, die Sie mit den meisten Malprogrammen bzw. dem Snapshot-Utility, das zu 1st Word gehört, erzeugen können, oder mit Zeichnungen aus objektorientierten Zeichenprogrammen wie GEM Draw oder Easy Draw füllen. Texte können Sie sowohl im Publisher selbst eingeben, was natürlich im allgemeinen nicht so schnell geht, wie mit einem einfachen Textprogramm, oder auch mit einer Textverarbeitung schreiben und dann in das Layout-Programm einladen.Dabei werden die Formate der Programme 1st Word plus, Word Writer PC, Word Writer ST, WordStar und Word Perfect sowie ASCII-Dateienproblemlos verstanden.

Bilder können im Format von GEM Paint, GEM Scan, Degas, Neochrome, PC Paintbrush oder Publishers Paintbrush geladen werden. Primitive Nachbearbeitungsmöglichkeiten stehen zur Verfügung (Lupe und

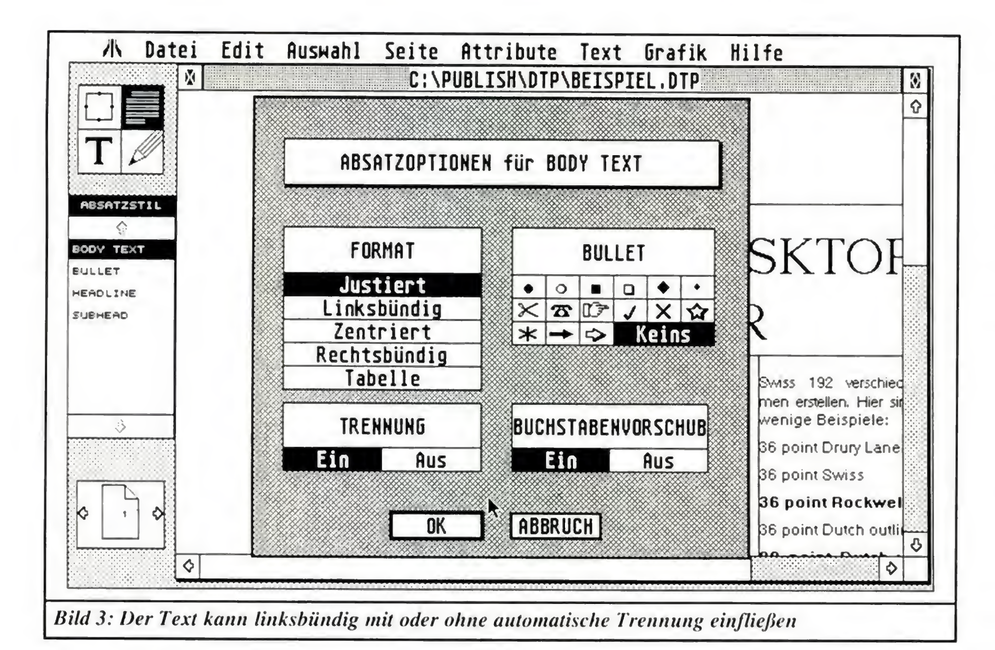


Bild 3: Der Text kann linksbündig mit oder ohne automatische Trennung einfließen

Skalieren / Ausschnitt bestimmen).

Auch Zeichnungen können in verschiedenen Formaten gelesen werden: GEM Draw und GEM Graph, Easy Draw und das Lotus 123-Format sind dem Publisher verständlich. Diverse Vektorgrafik-Funktionen sind zusätzlich eingebaut, so daß man einfache Illustrationen, Tabellen und Grafiken auch direkt im Timeworks-Programm zurechtzimmern kann. Nach dem Laden erscheint ein Desktop, der aus einer reichlich vollen Menüleiste, einem Layout-Fenster und einigen Bedienungselementen am linken Bildrand besteht. Mit diesen Bedienungselementen werden die wichtigsten Funktionen und Arbeitsmodi des Programms ausgewählt. Trotzdem kann das Layout-Fenster sie überdecken, wenn Sie einmal mehr von Ihrem Dokument sehen wollen.

Vier Arbeitsmodi gibt es im Programm: Im Rahmenmodus können Sie Rahmen zeichnen. Die anderen Modi dienen zur Bearbeitung des Rahmeninhalts: Textmodus und Grafikmodus dienen dazu, Texte bzw. Grafiken in einem Rahmen zu erstellen bzw. zu bearbeiten, der Absatzmodus dient dazu, das Absatzformat eines Textabsatzes zu verändern, oder die Absatzformate selbst zu edieren.

Unterhalb der vier Symbole für die Arbeitsmodi gibt es ein fensterähnliches Feld in dem, je nach Modus, entweder die Namen von eingeladenen Texten oder Bildern, die möglichen Textstile, die Grafikwerkzeuge oder die Namen der Absatzformate angezeigt werden. Durch Mausklick kann man die entsprechenden Felder anwählen. Am unteren Rand wird angezeigt, welche Seite sich gerade in der Bearbeitung befindet. Mit zwei Pfeilen kann im Dokument vor- oder zurückgeblättert werden.

Seitenlayout

Ein Dokument kann aus beliebig vielen Seiten bestehen, die Anzahl ist also nur vom Speicherplatz des Rechners abhängig. Zusätzlich gibt es sogenannte 'Master'-Seiten, auf denen man Layout-Elemente definieren kann, die auf mehreren Seiten des Dokuments vorkommen sollen. Man kann Layouts definieren und dann abspeichern, so daß man mühelos mehrere Dokumente im gleichen Layout erstellen kann.

Leider stehen nur vier verschiedene Seitenformate zur Verfügung: Das deutsche DIN A4 Format und die amerikanischen 'Letter',- 'Legal'- und 'Note'- Formate. Dabei kann man das Format, wenn es einmal festgelegt ist, auch nicht mehr ändern. Eine Möglichkeit, freie Formate zu bestimmen, fehlt hier doch unbedingt. Ein Dokument kann ein- oder doppelseitig sein. Doppelseitige Dokumente erlauben es, für gegenüberliegende linke und rechte Seiten jeweils ein eigenes Layout zu definieren, weil ja z.B. bei Büchern die gegenüberliegenden Seiten meistens nicht ganz gleich gestaltet werden. Rahmen können entweder auf den Master-Seiten definiert und dann automatisch auf jede neu erzeugte Seite übernommen, oder auch auf jeder Seite per Hand erzeugt werden. Um Rahmen genau positionieren zu können, gibt es sogenannte Spaltenleitern. Das sind sozusagen Pseudorahmen, eine Art Schablone für Rahmen. Man kann definieren, wieviele Spaltenleitern pro Seite es geben soll, wie groß der Abstand der Spaltenleitern von den vier Seitenrändern sowie der Abstand der Spaltenleitern untereinander sein darf. Mit einer Einrast-Funktion, ist es dann sehr einfach, Spalten exakt über den Spaltenleitern zu positionieren. Für Überschriften oder ausgefallene Positionen gibt es auch ein Hilfsmittel: Man kann sich exakte Größe und Position jedes Rahmens in einer Dialogbox anzeigen lassen und die Werte dort auch verändern.

Selbstverständlich gehören zu einem Seitenlayout auch Kopf- und Fußzeilen, die hier für rechte und linke Seiten getrennt definiert werden können.

Textbearbeitung

Die wohl häufigste Tätigkeit in einem DTP-Programm hat mit dem Layout von Texten zu tun. Die wichtigsten Funktio-

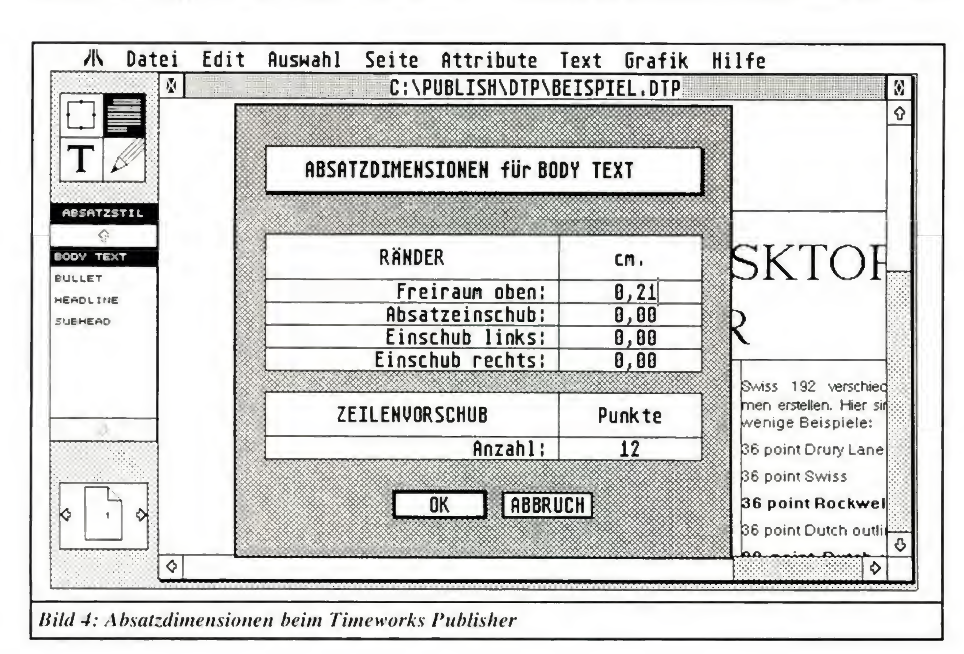


Bild 4: Absatzdimensionen beim Timeworks Publisher

nen des Programms haben also mit den Design-Möglichkeiten für Texte zu tun. Hierbei wiederum das Wichtigste sind die Zeichensätze. Im Moment stehen 5 Zeichensätze zur Verfügung. Davon sind 2, Swiss und Dutch (Times-ähnlich), in Größen zwischen 7 und 72 Punkt vorhanden, Rockface nur in 20 und 40 Punkten, ein weiterer Schrifttyp, der nur aus gestylten Großbuchstaben besteht, in 20 bis 40 Punkten, sowie ein Sonderzeichensatz mit ungefähr 20 Symbolen in Größen zwischen 7 und 70 Punkten. Jeder der Fonts enthält auch die internationalen Sonderzeichen. Diese Schriften können jeweils in einer ganzen Reihe von Textstilen dargestellt werden: fett, kursiv, unterstrichen, hell, invers, umrandet und hoch- und tiefgestellt.

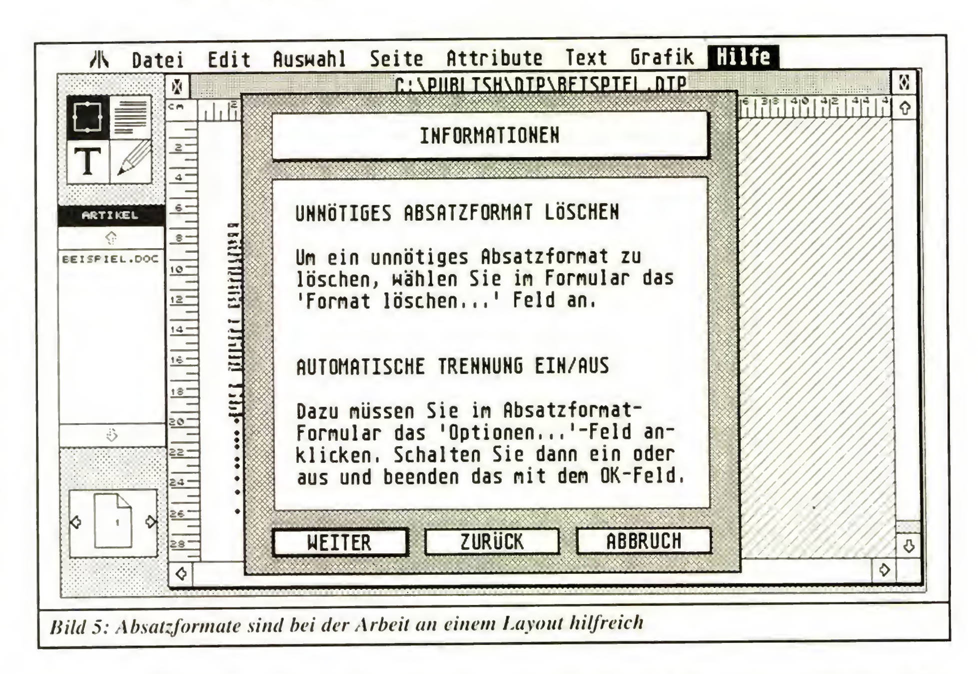


Bild 5: Absatzformate sind bei der Arbeit an einem Layout hilfreich

Bei der Übernahme von Texten aus Textverarbeitungen bleiben die dort definierten Textattribute übrigens im allgemeinen erhalten. Selbst Fußnoten aus 1st Word bleiben erhalten, müssen allerdings oft neu formatiert werden. Jeder Textabsatz kann mit einem eigenen Textformat ausgestattet werden. Dieses Absatzformat beinhaltet Informationen über die Textformatierung (Blocksatz, Zentriert, Rechts- oder Linksbündig), Einrückungen, Zeilenabstand, Abstände vom Rahmenrand oder dem letzten Abstand usw. Selbst Schrifttyp und Größe lassen sich vorprogrammieren, obwohl diese Parameter auch innerhalb eines Absatzes beliebig ändern lassen.

Für Einrückungen läßt sich ein spezielles Sonderzeichen definieren, das immer am Anfang des Absatzes gedruckt wird, z.B. ein dicker Punkt. Dies ist sehr hübsch, um beispielsweise in einem Katalog einzelne Punkte aufzulisten und dabei deutlich voneinander abzuheben, ohne aber jedesmal 'per Hand' ein Zeichen aus einem Spezialfont einsetzen zu müssen.

Ebenfalls zum Absatzformat gehören Informationen über die automatische Trennung. Damit wäre bereits verraten, daß eine Trennhilfe im Programm eingebaut ist, die auch recht gut funktioniert. Selbstverständlich kann man die Vorschläge der automatischen Trennung auch jederzeit nachträglich per Hand verändern oder die Trennung ganz zurücknehmen.

Schließlich kann man auch Tabulatoren setzen, wobei mehrere Tabulatorentypen wie Dezimal- oder zentrierter Tabulator zur Verfügung stehen.

Absatzformate lassen sich auf Funktionstasten legen, so daß man für den Aufruf nur eine Funktionstaste zu drücken braucht. Die Textformatierung ist mit verschiedenen Parametern steuerbar: Man kann den maximalen und minimalen Abstand zwischen zwei Worten, der bei der Blockformatierung entstehen darf, setzen, sowie eine Zone am Zeilenende definieren, in der die Trennung aktiv ist. Bei der Blockformatierung kann man wählen, ob Leerräume außer zwischen Worten auch zwischen den Buchstaben einzelner Worte eingefügt werden sollen, wodurch die Leerzeichen zwischen den Wörtern kleiner werden, dafür aber die Schrift leicht gesperrt wirkt. Dieses automatische 'Kerning' kann jederzeit durch eine manuelle Kerning-Funktion verändert werden: Der Abstand zwischen beliebigen Buchstaben läßt sich in einer Dialogbox verändert werden.

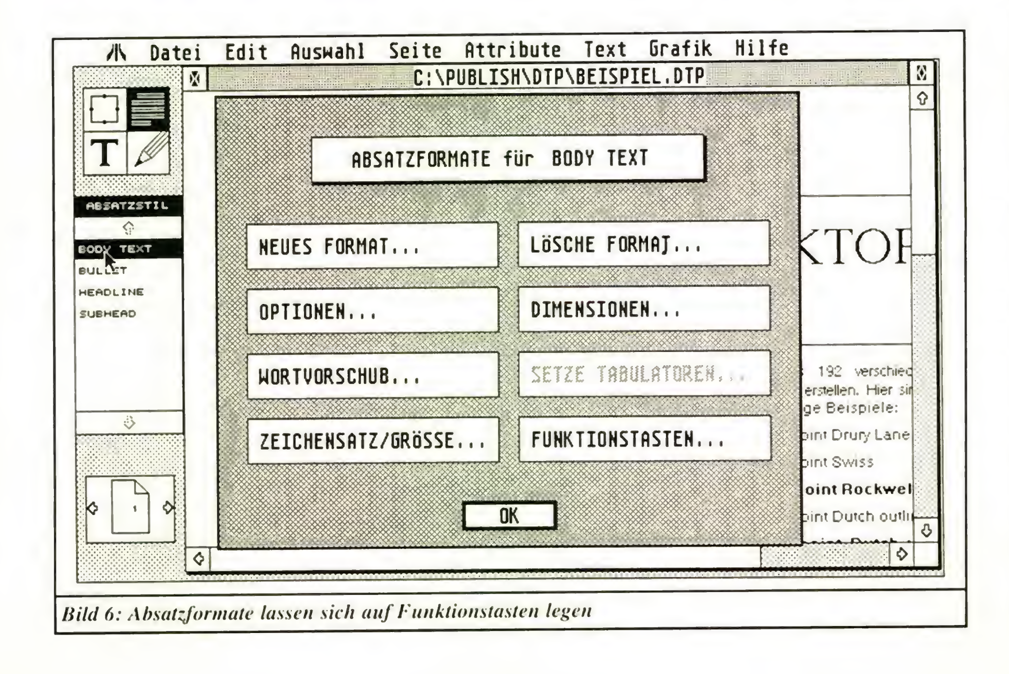


Bild 6: Absatzformate lassen sich auf Funktionstasten legen

Die Absatzformate helfen bei der Arbeit an einem Layout sehr, besonders, wenn viele Seiten in einem durchgängigen Stil gestaltet werden müssen. Man kann sich dann zum Beispiel ein Absatzformat für Überschriften, eines für Unterüberschriften, eines für den eigentlichen Text, eines für Registerseiten usw. konstruieren. Bevor man dann beginnt, den Rahmen mit dem dafür vorgesehenen Text zu füllen, braucht man nur das gewünschte Absatzformat zu wählen und schon erhält der Text das entsprechende Styling. Damit muß man nicht jedesmal alle Parameter neu einstellen. Vorbereitete Parameter anwählen, schon fertig. Auch wenn Sie öfter ähnliche Druckschriften erstellen müssen, hilft diese Möglichkeit: Sie können die Absatzformate abspeichern und dann für weitere Dokumente verwenden. Ansonsten stellt der Timeworks Publisher noch die üblichen Text-Funktionen Suchen und Ersetzen zur Verfügung. Beliebige Textblöcke können ausge-

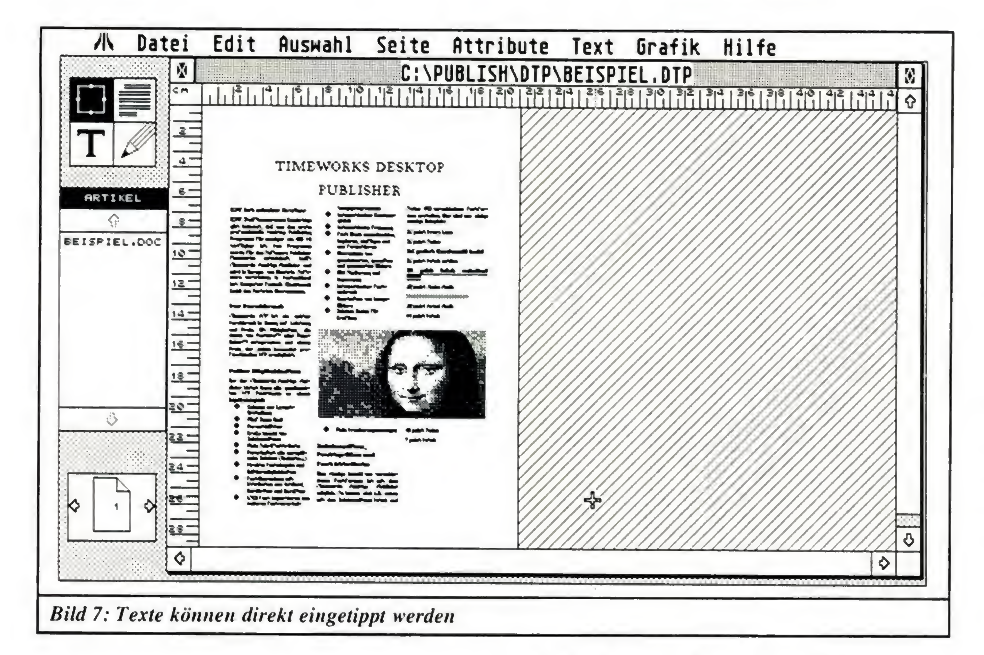


Bild 7: Texte können direkt eingetippt werden

schnitten oder kopiert und wieder eingeklebt werden.

Edieren

Wie bereits erwähnt, kann Text direkt im Publisher eingetippt werden. Da die gesamte Formatierung aber sofort bei der Eingabe erfolgt, ist die Arbeit damit etwas langsam. Mir erscheint es allemal effizienter, Text mit einer Textverarbeitung einzugeben und nur kleine Änderungen im Layout-Programm vorzunehmen.

Text kann übrigens nur in der normalen Leserichtung von links nach rechts verwendet werden. Vertikaler oder schräger Text ist leider nicht möglich. Längere Texte werden im allgemeinen nicht in einen Rahmen passen. Daher ist es möglich, einen Text nacheinander in mehrere Rahmen fließen zu lassen. Diese Rahmen hängen dann in einer Kette aneinander, so daß sie sich bei jeder Textarbeit wie ein Rahmen verhalten. Wenn man also zum Beispiel die Rahmengröße verändert oder zusätzlichen Text einfügt, wirkt sich das automatisch auf alle verbundenen Rahmen aus; man muß nicht den ganzen Text per Hand neu auf die Rahmen verteilen.

Rahmen können auch über andere Rahmen gezeichnet werden. Dabei kann man wählen, ob der obere Rahmen Text im unteren Rahmen überdecken soll, oder ob der Text um diesen überdeckenden Rahmen herumfließen soll. Leider kann Text auf diese Weise nur um eckige Rahmen fließen, nicht um beliebig geformte Objekte. Überdeckende Rahmen können auch durchsichtig sein. Außerdem kann man einen verdeckten Rahmen jederzeit mit einem besonderen Kommando nach vorne holen.

Zeichenkünstler?

Aus eingeladenen Pixelgrafiken können beliebige Ausschnitte herausgeschnitten werden. Auch ist es möglich, diese Ausschnitte frei zu skalieren, wobei die Proportionen total verfremdet oder auch automatisch erhalten bleiben können. Eine Lupen-Funktion erlaubt es, jeden einzelnen Bildpunkt einzeln zu bearbeiten.

Mehr Möglichkeiten gibt es im Bereich Vektorgrafik. Importierte Vektorgrafiken können mit diesen Funktionen allerdings leider nicht nachbearbeitet werden. Sieben Zeichenfunktionen stehen zur Verfügung: Linien, Rechtecke mit eckigen oder runden Ecken, Kreise, Ellipsen, Freihandzeichnungen und Polygone. Jede Figur kann in verschiedenen Strichstärken gezeichnet werden, auch das Aussehen des Linienendes kann verändert werden (z.B. Pfeilspitzen). Als Edierhilfe gibt es ein einstellbares Punktraster. Auch Füllmuster gibt es; auf insgesamt 36 verschiedene Arten kann eine Fläche gefüllt werden. Da es sich um Vektorgrafik handelt, kann jedes Element der Zeichnung nachträglich beliebig manipuliert werden. Selbstverständlich kann man Grafikobjekte genauso ausschneiden, kopieren und einfügen wie Texte.

Bedienung

Das Programm, selbstverständlich voll mausgesteuert, ist leicht zu bedienen. Für viele wichtige Funktionen gibt es Tastaturkommandos. Der Timeworks Publisher ist auch schnell genug, damit man nicht bei der Arbeit pausenlos darauf warten muß, daß irgendetwas endlich auf dem Bildschirm erscheint.

Anfänger können sich durch eine umfangreiche Hilfe-Funktion unterstützen lassen. In der Menüleiste wählt man das Thema, über das man informiert werden will, aus. Es erscheint eine Dialogbox, in der man im Hilfstext vor- und zurückblättern kann, bis man die heiß ersehnte Hilfestellung gefunden hat.

Für das Layout gibt es ebenfalls einige Hilfen: Die Bemaßung des Dokuments kann in Picas und Punkten, Zentimetern oder Zoll (1/8tel oder 1/10tel Teilung) dargestellt werden, ein Lineal kann man sich auch einblenden lassen, damit man immer die richtigen Stellen im Dokument trifft. Da der Atari-Bildschirm leider zu klein ist, um eine ganze Seite gleichzeitig auf den Bildschirm zu bekommen, gibt es verschiedene Verkleinerungsmöglichkeiten. So kann man sich bis zu zwei Seiten in Ganzseitendarstellung auf dem

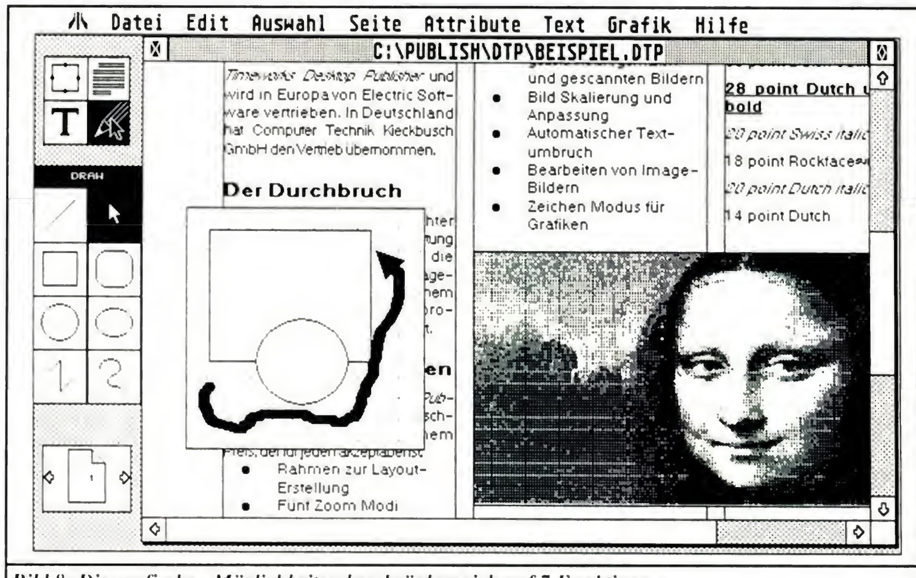
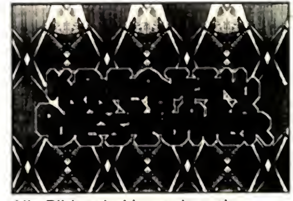


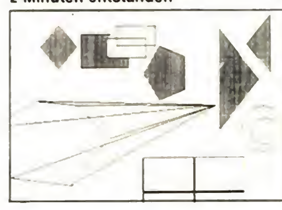
Bild 8: Die grafischen Möglichkeiten beschränken sich auf 7 Funktionen

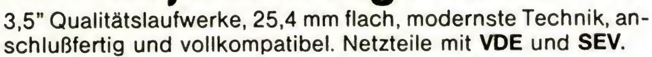


Alle Bilder sind in weniger als



2 Minuten entstanden





Bildschirm zeigen lasssen. Auch halbe oder doppelte Originalgröße ist möglich. Mir persönlich fehlt eine Funktion, die das Dokument genau soweit verkleinert, daß eine Dokumentbreite auf den Bildschirm paßt.

Druck

Es gibt eine ganze Menge Druckertreiber, für 9- und 24-Nadeldrucker sowie verschiedene Laserdrucker, von Atari bis Hewlett Packard. Auch ein PostScript-Treiber ist vorhanden. Die Druckqualität auf 24-Nadel- und Laserdruckern ist gut, allerdings wird das gleiche Dokument je nach Druckeranpassung mit unterschiedlichen Schriftgrößen gedruckt! Dies sollte sich dringend ändern. Eine 12-Punkt-Times sieht mit einem 24-Nadler eher nach 9-Punkt aus, während sie auf einem 9-Nadel- oder Laserdrucker in etwa die gleiche Größe aufweist. An unseren Textproben sehen Sie, daß in den beiden Ausdrucken (von 24-Nadel- und Laserdrucker) kaum mehr Punkte bei der Schrift in Normalgröße zu sehen sind, lediglich die Überschriften weisen noch Eckchen auf.

Die 9-Nadel-Qualität entspricht ungefähr dem, was man auch von Malprogrammen, die mit GDOS-Fonts arbeiten, gewöhnt ist. Dafür geht der Ausdruck auf diese Weise ziemlich flott vonstatten, mit einem 24-Nadel-Drucker dauert es nämlich ewig, bis eine Seite gedruckt ist. Mit dem Atari-Laser geht es sehr schnell, allerdings mußten wir in unserem Test das Programm manchmal sehr häufig auffordern, bevor es sich dazu herabließ, eine Seite an den Drucker zu übergeben.

TIMEWORKS DESKTOP PUBLISHER

GST ist wieder Erster

GST Software aus Cambridge gibt bekannt, daß nun das erste professionelle Desktop Publishing Programm für weniger als 400 DM verfügbar ist. Das Programm wurde für den Software Publisher Timeworks entwickelt, heißt *Timeworks Desktop Publisher* und wird in Europa von Electric Software vertrieben. In Deutschland hat Computer Technik Kieckbusch GmbH den Vertrieb übernommen.

Der Durchbruch

Timeworks DTP ist ein echter Durchbruch in Bezug auf Leistung und Preis. Mit Fähigkeiten, die denen von Ventura™ oder PageMaker™ entsprechen, und einem Preis, der jedem Anwender professionalles DTP ermöglicht.

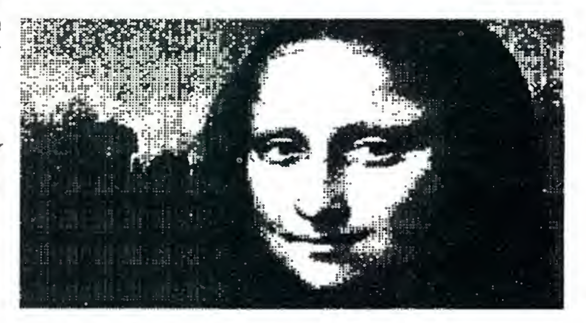
Voller Möglichkeiten

Nur der *Timeworks Desktop Publisher* bietet Ihnen alle gewünschten DTP Funktionen zu einem Preis, der für jeden akzeptabel ist

- Rahmen zur Layout-Erstellung
- Fünf Zoom Modi
- Formatblätter
- Große Anzahl von Zeichensätzen
- Viele Schriftattribute
- Verarbeitet alle europäische Zeichen (Umlaute...)
- Direkte Texteingabe und Editiermöglichkeiten
- Textübernahme mit Attributen aus 1stWord, WordWriter und WordStar
- ASCII Text importieren aus anderen Textverarbeitungsprogrammen
- Automatischer Randausgleich
- Automatische Trennung
- Text Block ausschneiden, kopieren, einfügen und neu formatieren
- Übernahme von gezeichneten, gemalten und gescannten Bildern
- Bild Skalierung und Anpassung
- Automatischer Textumbruch
- Bearbeiten von Image-Bildern
- Zeichen Modus für Grafiken
- Viele Druckeranpassungen

Zeichensätze, Punktgrößen und Text Attribute

Eine riesige Anzahl von verschiedenen Textformen ist mit dem *Timeworks Desktop Publisher* möglich. So lassen sich z.B. schon mit den Zeichensätzen Dutch und Swiss 192 verschiedene Textformen erstellen. Hier sind nur einige wenige Beispiele:



36 point Drury Lane

36 point Swiss

36 point Rockwell bold

36 point Dutch outline

28 point Dutch underlined bold

20 point Swiss italic

18 point Rockface superscript

20 point Dutch italic

14 point Dutch

10 point Swiss

7 point Dutch

Bild 9: Ein fertiger Ausdruck mit einem 24-Nadel-Drucker

Fazit

Ich möchte den Timeworks Desktop Publisher mit dem Publishing Partner vergleichen, der ja nun schon seit längerem auf dem Markt ist. In diesem Vergleich stellt sich heraus, daß keines der beiden Programme herausragend besser ist. Mir persönlich gefällt die Bedienerführung des Publishing Partners besser, auch ist er in manchen Punkten (z.B. mehrere Druckertreiber gleichzeitig, Dokumentformate) einfach flexibler. Dafür ist die Druckqualität mit Matrixdruckern recht bescheiden.

Besonders schön beim Timeworks Publisher ist die Idee des Absatzformates. Damit kann man Texte wirklich schnell 'in Form' bringen. Auch die Trennhilfe funktioniert sauber und gut, was man von der des Publishing Partners nicht unbedingt behaupten kann. Übrigens verwendet die Trennhilfe ein ASCII-Lexikon, das den Eindruck macht, als könnte man es leicht selbst ergänzen. Die Grafikmöglichkeiten sind gut und leicht zu handhaben. Gerade Geschäftsgrafiken oder andere 'einfache' Illustrationen und Diagramme lassen sich schnell und leicht erstellen, ohne daß erst ein spezielles Grafikprogramm benötigt wird. Gegenüber Grafiken aus Malprogrammen hat die Vektorgrafik den Vorteil, daß sie auf Druckern mit höherer Auflösung auch in besserer Qualität wiedergegeben werden kann.

Ein paar mehr Fonts sollte Timeworks zur Verfügung stellen. Zwei vollständige Schriftsätze sind für einen professionellen Einsatz eines derartigen Programmes einfach nicht genug. Zumindest die weitverbreitetsten Standardschriften müssen einfach vorhanden sein.

Störend finde ich am Timeworks Publisher: Die mangelnde Flexibilität bei der Formatwahl und die Tatsache, daß das Programm bei einem Druckerwechsel komplett neu installiert werden muß. Die Betriebssicherheit ist brauchbar. Abstürze gibt es sehr selten, auch skurille Ergebnisse kommen nicht oft vor. Lediglich Texteingaben auf der Masterseite führen manchmal zu ungewollten Resultaten.

Ein paar Dinge, die mir fehlen: Auch (oder besser gerade) ein DTP-Programm sollte Fußnoten beherrschen, ebenso automatische Inhalts- und Stichwortverzeichnisse unterstützen. Auch etwas mehr Flexibilität bei der Schreibrichtung (vertikal, schräg) würde mich nicht stören.

Alles in allem ist der Timeworks Desktop Publisher ein Programm, das vernünftige Ergebnisse mit verhältnismäßig kleinem Aufwand erlaubt. Allzu ausgeflippte

TIMEWORKS DESKTOP PUBLISHER

GST ist wieder Erster

GST Software aus Cambridge gibt bekannt, daß nun das erste professionelle Desktop Publishing Programm für weniger als 400 DM verfügbar ist. Das Programm wurde für den Software Publisher Timeworks entwickelt, heißt *Timeworks Desktop Publisher* und wird in Europa von Electric Software vertrieben. In Deutschland hat Computer Technik Kieckbusch GmbH den Vertrieb übernommen.

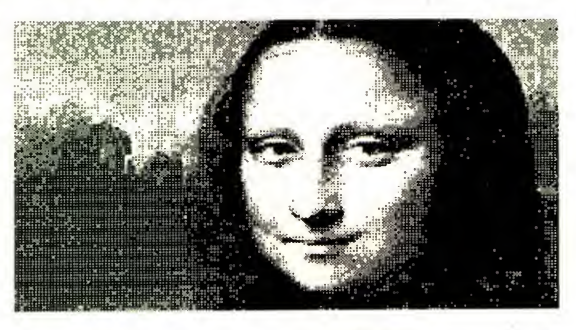
Der Durchbruch

Timeworks DTP ist ein echter Durchbruch in Bezug auf Leistung und Preis. Mit Fähigkeiten, die denen von Ventura™ oder PageMaker™ entsprechen, und einem Preis, der jedem Anwender professionalles DTP ermöglicht.

Voller Möglichkeiten

Nur der *Timeworks Desktop Publisher* bietet Ihnen alle gewünschten DTP Funktionen zu einem Preis, der für jeden akzeptabel ist.

- Rahmen zur Layout-Erstellung
- Fünf Zoom Modi
- Formatblätter
- Große Anzahl von Zeichensätzen
- Viele Schriftattribute
- Verarbeitet alle europäische Zeichen (Umlaute...)
- Direkte Texteingabe und Editiermöglichkeiten
- Textübernahme mit Attributen aus 1stWord, WordWriter und WordStar
- ASCII Text importieren aus anderen Textverarbeitungsprogrammen
- Automatischer Randausgleich
- Automatische Trennung
- Text Block ausschneiden, kopieren, einfügen und neu formatieren
- Übernahme von gezeichneten, gemalten und gescannten Bildern
- Bild Skalierung und Anpassung
- Automatischer Textumbruch
- Bearbeiten von Image-Bildern
- Zeichen Modus für Grafiken
- Viele Druckeranpassungen



Zeichensätze, Punktgrößen und Text Attribute

Eine riesige Anzahl von verschiedenen Textformen ist mit dem *Timeworks Desktop Publisher* möglich. So lassen sich z.B. schon mit den Zeichensätzen Dutch und Swiss 192 verschiedene Textformen erstellen. Hier sind nur einige wenige Beispiele:

36 point Drury Lane

36 point Swiss

36 point Rockwell bold

36 point Dutch outline

28 point Dutch underlined bold

20 point Swiss italic

18 point Rockface superscript

20 point Dutch italic

14 point Dutch

10 point Swiss

7 point Dutch

Bild 10: Der gleiche Ausdruck mit dem ATARI Laserdrucker

Möglichkeiten bietet das Programm nicht, aber das muß für die meisten Anwendungen ja auch wirklich nicht sein. Die Druckqualität ist gut. Im Vergleich mit dem direkten Konkurrenten Publishing Partner fällt allerdings der höhere Preis ins Gewicht: Der Timeworks Desktop Publisher kostet 398.- DM, der Publishing Partner nur 249.- DM. Dafür bietet der Publishing Partner bei vergleichbarer Qualität des Laserausdrucks mehr Schrifttypen sowie (rein subjektiv aus meiner Sicht) eine angenehmere Bedienung, was nicht heißen soll, daß der Timeworks Publisher schlecht zu bedienen wäre. Es ist eher ein Unterschied zwischen 'gut' und 'besser'. Andererseits ist bei Timeworks die Idee mit den Absatzformaten wirklich bestechend einfach und nützlich in der Anwendung, die Trennhilfe des Timeworks Publishers funktioniert sauberer. Auf jeden Fall eine schwere Entscheidung. Vielleicht entschließt man sich bei Timeworks ja noch, die ein oder andere Schwäche (vor allem, bitte, mehr Fonts und größere Ausdruck-Flexibilität !) zu beheben. Eine schlechte Wahl ist der Timeworks Publisher auf keinen Fall. Wie für alle momentan erhältlichen DTP-Programme, egal für welchen Rechnertyp, gilt auch hier: Erwarten Sie keine Wunderdinge. Eine ganze Zeitung z.B. professionell mit einem DTP-Programm zu layouten ist immer noch unbefriedigend. Wenn Ihre Projekte aber nicht ganz so groß sind, kann Ihnen der Timeworks Publisher eine große Hilfe sein.

CS

Vertrieb:
Computer Technik Kieckbusch GmbH
Baumstammhaus
5419 Vielbach
Tel: 02626/78336


vortex
COMPUTERSYSTEME
TEST-WOCHEN
Mit vortex 1 Woche nach New York!
...TESTEN,
VERGLEICHEN,
GEWINNEN!
vortex card
vortex HDplus 20

DIE VORTEX-TESTWOCHEN WERDEN IHREN COMPUTER-ALLTAG VERÄNDERN

Sie erleben die nahezu unbegrenzten Möglichkeiten der vortex-Computertechnik. Sie werden erleben, wie ein „normaler“ Personal-Computer plötzlich zum Kraftpaket wird.

Und Sie können gewinnen. Durch Ihr Test-Urteil. Über Leistung, Preis und Handling. Jede Testkarte nimmt an der wöchentlichen Sachpreisverlosung und an der Verlosung einer Flugreise nach New York teil. Viel Erfolg.

DIE VORTEX TESTWOCHEN: NUR BEI DIESEN VORTEX-FACHHANDELS-PARTNERN:



1000 Berlin 31 · Data Play
Bundesallee 25 · ☎ 8 61 33 15

1000 Berlin 61
Hifi Computerstudio Schlichting
Katzbachstraße 8 · ☎ 7 86 43 40

2000 Hamburg
Bit Computershop
Osterstraße 173 · ☎ 49 44 00

2000 Hamburg 71
Createam Hard + Software
Bramfelder Chaussee 300
☎ 6 41 50 91

2000 Hamburg · G M A mbH
Wandsbecker Chaussee 58
☎ 2 51 24 16

2000 Hamburg 1 · Uwe Putscher
Computer & Zubehör
Berliner Tor 3 · ☎ 24 98 19

2300 Kiel
MCC Microcomputer Christ
Dreiecksplatz 7 · ☎ 56 70 41

2800 Bremen · PS Data
Faulenstraße 48–52 · ☎ 17 05 77

3000 Hannover 1 · Com Data
Schiffgraben 19 · ☎ 32 67 36

3000 Hannover
trendData Computer GmbH
Am Marstall 18–22 · ☎ 16 60 50

3000 Hannover 71
PCH-Computer GmbH
Großer Hillen · ☎ 52 25 79

3100 Celle
Computertechnik Center Haupt
Hannoversche-Heer-Straße 68
☎ 88 15 02

3170 Gifhorn
Computerhaus Gifhorn
Pommernring 38 · ☎ 5 44 98

3200 Hildesheim
trendData Computer GmbH
Zingel 15 · ☎ 3 90 94

3300 Braunschweig
Computerstudio Braunschweig
Rebenring 49–50 · ☎ 33 32 77–79

3550 Marburg
Computerstudio Lemmer
Augustiner Gasse 4 · ☎ 2 75 66

4300 Essen 1 · Haase Computer
Wiedfeldstraße 77 · ☎ 42 25 75

4300 Essen · Winkler & Müller
Lindenallee 59–67 · ☎ 8 10 20-0

4800 Bielefeld · Microtec mbH
Carl-Severing-Str. 190 · ☎ 45 99-150

4950 Minden
PCM-Computer GmbH i. G.
Obermarktstraße 21

4972 Löhne
Radio Fritz Obermeier
Am Hauptbahnhof + Bünder Str. 20
☎ 32 46

5000 Köln 41 · AB Computer
Wildenburger Straße 21
☎ 4 30 14 42

5000 Köln · Büromaschinen Braun
Richard-Wagner-Str. 39 · ☎ 21 91 71

5000 Köln 1
HSS Harmel Scollar Schmitthals
Darmstädter Straße 20 · ☎ 31 62 07

5120 Herzogenrath
Roda Soft EDV Fachgeschäft
Bahnhofstraße 6 · ☎ 7 91 00

5210 Troisdorf · Logi Team
Kölner Straße 132 · ☎ 7 18 97

5300 Bonn · EDV Vertriebsgesellschaft mbH Hansen & Gieraths
Münsterstraße 1 · ☎ 72 90 80

6000 Frankfurt 90 · Data Technics
Elbinger Straße 9 · ☎ 70 66 94

6000 Frankfurt
Waizenegger GmbH & Co. KG
Kaiserstraße 41 · ☎ 27 30 60

6090 Rüsselsheim · Büro Emig
Haßlocher Straße 26 · ☎ 6 10 71

6100 Darmstadt-Eberstadt
Heim Computertechnik
Heidelberger Landstraße 194
☎ 5 60 57

6330 Gießen
Computerstudio Lemmer
Mäusburg 4 · ☎ 39 02 93

6600 Saarbrücken
W. N. Pfeiffer EDV-Abteilung
Beethovenstraße 28 / Karcherstraße
☎ 3 22 11

6680 Neunkirchen
Computer-Profi-Center Shop 64
Lutherstraße 7 · ☎ 2 37 13

6700 Ludwigshafen · Microcomp. Vertriebsgesellschaft mbH MKV
Im Bismarckzentrum (BZ)
Bismarckstraße 71
☎ 52 54 95

6720 Speyer
Etzkorn Computer
Auestraße 20 · ☎ 3 24 28

7000 Stuttgart 1
Schreiber Computer
Rotebühlplatz 10
☎ 22 70 99

7000 Stuttgart 1
Schreiber Computer
Breuninger Subway /
Marktstr. 1–3 · ☎ 22 19 97

7032 Sindelfingen
Schreiber Computer
Breuningerland Sindelfingen
Tilsitterstraße 15 · ☎ 8 22 59

7410 Reutlingen
Brock Computer
Untere Gerber Straße 15 · ☎ 3 42 87

7520 Bruchsal · Jöst Computer
Werner-von-Siemens-Straße 47
☎ 8 50 91–96

7530 Pforzheim
Schreiber Computer
Marktplatz 4 · ☎ 35 66 99

7602 Oberkirch · Data Systeme
Radio Elektro Kuderer GmbH
Hauptstraße 79 · ☎ 41 14

7750 Konstanz
Computertechnik Rössler
Rheingutstraße 1 · ☎ 2 18 32

7800 Freiburg · Pyramid Computer
Kartäuser Straße 59 · ☎ 3 15 71

7890 Waldshut 1 · Hettler GmbH
Lenzburger Straße 4 · ☎ 30 94

7980 Ravensburg · CSE Schauties
Schussenstr. 4 · ☎ 2 61 38 + 2 64 97

7990 Friedrichshafen
Müller Elektronik
Am Anger 1 · ☎ 5 46 93

8000 München 80
ABAC München
Kellerstraße 11 · ☎ 4 48 99 88

8000 München
Ludwig Computer & Bürotechnik
Ingolstädter Straße 62 L
☎ 3 11 30 66

8000 München 2 · Seemüller
GmbH Computer & Bürobedarf
Schillerstraße 18 · ☎ 59 42 81

8032 Gräfelfing · Pro Markt
Pasinger Straße 94 · ☎ 85 10 43

8150 Holzkirchen
Münzenloher GmbH
Tölzer Straße 5 · ☎ 18 13

8170 Bad Tölz
Elektronic Center Uwe Langheinrich
Wachter Straße 13 · ☎ 4 15 65

8263 Burghausen
Heinz Jaskulski Büromaschinen
Mautner Straße 243 · ☎ 6 33 20

8500 Nürnberg 80
CG Computerstore GmbH
Hochstraße 11 · ☎ 28 90 28

8500 Nürnberg 80
Technoland Nürnberg
Rothenburger Str. 451 · ☎ 6 58 88 03

8650 Kulmbach · Schneider
Büroelektronik und -service
Obere Stadt 36 · ☎ 16 17

8700 Würzburg
Duttenhofer GmbH + Co. KG
Domstraße 5 · ☎ 5 02 15

8720 Schweinfurt · Uhlenhut
Albrecht-Dürer-Platz 2 · ☎ 65 20

8850 Donauwörth
SBC Hans Seefried KG
Bahnhofstraße 3 · ☎ 49 10

8900 Augsburg
Fichtinger + Seeger
Bahnhofstraße 15 · ☎ 51 80 09



4102 Binningen · Computer Trend
Kronenplatz/Paradiesstraße 2
☎ 47 88 64

5000 Aarau · Computer Trend
Kasernenstraße 26 · ☎ 22 63 33

5430 Wettingen · Computer Trend
Zentralstraße 93 · ☎ 27 16 60

8021 Zürich · Computer Trend AG
Langstraße 31 · ☎ 2 41 73 73

8400 Winterthur
Computer Trend
St. Galler Straße 41 · ☎ 27 96 96

ACHTEN SIE AUF DIE TESTWOCHEN-TERMINE IN IHRER LOKALZEITUNG



vortex Computersysteme GmbH · Falterstraße 51–53 · D-7101 Flein · Telefon (0 71 31) 5 20 61
Computersysteme vortex AG · Bundesplatz 3 · CH-6300 Zug · Telefon (0 42) 21 84 42



AUF DER SCHWELLE ZUM LICHT

Directory-Verwaltung Teil 2

Heute geht es um die Anwendung der im ersten Teil besprochenen Grundlagen der Directory-Verwaltung. Hauptsächlich handelt es sich dabei um das Suchen (und hoffentlich auch Finden) von Dateien, einem der Kernstücke von GEMDOS. Für diese Aufgabe werden die wichtigen Routinen 'd_srcname' und 'd_getdir' (Namen sind wie üblich frei erfunden, Ähnlichkeiten mit lebenden Routinen wären rein zufällig) benötigt, die schon im ersten Teil erwähnt wurden. Bevor wir ans Eingemachte gehen, möchte ich aber noch einige allgemeine Dinge vorausschicken.

Ein Name - zwei Teile

GEMDOS-Dateinamen bestehen aus zwei Teilen: Der Hauptname besteht aus bis zu 8 Zeichen, die Namenserweiterung (Extender) aus bis zu 3 Zeichen. Bei dem dem Benutzer bekannten Namensformat sind beide Teile durch einen '.' getrennt. Fehlt der Extender, so kann der '.' weggelassen werden (GEMDOS selbst macht dies bei "Rückgabe" von Dateinamen wie bei 'Dgetpath' und 'Fsfirst'/'Fsnext' immer).

Das GEMDOS-interne Format kennt nur genau 11 Zeichen lange Dateinamen. Zu kurze Namensteile werden mit Leerzeichen auf die 8 bzw. 3 Zeichen aufgefüllt. Daher ist auch kein Trennzeichen zwischen den beiden Teilen notwendig. Ein Beispiel: Die Datei 'DATEI.C' wird im internen Format durch 'DATEI   C ' repräsentiert.

Ersetzungszeichen

Bei der Spezifikation von Dateinamen, nach denen gesucht werden soll, können die Ersetzungszeichen (Wildcards) '*' und '?' verwendet werden.

Ein '?' zeigt an, daß an seiner Stelle ein beliebiges Zeichen stehen darf. So wird bei einem Suchnamen 'H?LLO' sowohl die Datei 'HALLO' als auch 'HELLO' gefunden.

Bei einem '*' sind alle weiteren Zeichen bis zum Ende des Namensteils, in dem das '*' vorkommt, beliebig. Der Suchname 'H*.C' läßt GEMDOS z.B. 'HALLO.C' oder 'HXXX.C', nicht aber 'HALLO.S' finden.

Weitere Zeichen nach dem '*' (bis zum Ende des Namensteils) werden ignoriert.

Intern wird übrigens das '*' in '?' umgewandelt, indem der betreffende Namensteil bis zu seiner maximalen Länge von 8 bzw. 3 Zeichen mit '?' aufgefüllt wird.

Die Ersetzungszeichen können immer dann verwendet werden, wenn nach einer schon vorhandenen Datei gesucht werden soll (z.B. bei 'Fopen' oder dem Quellnamen von 'Frename'). Alle GEMDOS-Funktionen außer 'Fsfirst'/'Fsnext' arbeiten jedoch immer nur auf einer Datei. Dies ist stets die im Directory am "weitesten vorne", also bei niedriger Dateiposition stehende.

Wenn alle Dateien, die das Suchkriterium erfüllen, angesprochen werden sollen, so muß dies entweder durch wiederholte Anwendung (z.B. bei 'Fdelete') oder durch 'Fsfirst'/'Fsnext' (s.u.) geschehen. Bei Dateipfaden darf nur der eigentliche Dateiname Ersetzungszeichen enthalten. Die Directories müssen immer vollständig angegeben werden.

Dateiattribut

Bisher habe ich mich immer um das Dateiattribut gedrückt, das schon des öfteren erwähnt wurde. Aber das dürfte wohl ganz im Sinne des GEMDOS liegen, da es

Bit	Bedeutung, wenn gesetzt	engl. Kurzbezeichnung
0	Datei ist schreibgeschützt	read only
1	Datei ist versteckt	hidden
2	Datei ist (versteckte) Systemdatei	system
3	Dateieintrag ist der Diskettenname	volume label
4	Datei ist ein Ordner	subdirectory
5	Datei ist "archiviert"	archive
6,7	reserviert (sollten gelöscht sein)	

Tab. 1: Das Dateiattribut

sich auch nicht sonderlich darum kümmert. Das Attribut ist nämlich zu weit weniger zu gebrauchen, als es im ersten Moment scheint. Jede in einem Directory verzeichnete Datei hat also ein sogenanntes Dateiattribut, durch das ihre Verwendung und bestimmte Eigenschaften charakterisiert werden. Das Attribut wird durch einen Byte-Wert repräsentiert, wobei bestimmten Bits eine Eigenschaft zugewiesen wird. In einigen Fällen ist ihre Kombination möglich. Die Bedeutung dieser Bits ist in Tab. 1 zusammengestellt.

Der 'read only'-Modus soll ein softwaremäßiger Schreibschutz einzelner Dateien sein. Er verhindert das Öffnen einer Datei mit 'Fopen' im Modus "schreiben" oder "lesen und schreiben". Weiterhin wird er bei 'Fdelete' und 'Fcreate' berücksichtigt.

'Fwrite' kümmert sich jedoch herzlich wenig um den Zugriffsmodus, so daß nur zum Lesen geöffnete Dateien trotzdem beschrieben werden können (s. Juni-Ausgabe).

Versteckte Dateien ('hidden' oder 'system') sollten normalerweise bei der Anzeige von Directories nicht erscheinen. Der Desktop berücksichtigt dies sogar, d.h. versteckte Dateien werden einfach nicht angezeigt (s. auch 'Suchattribut'). Der Begriff "System-Datei" hat weiter keine tiefschürfende Bedeutung, er klingt geheimnisvoller als er ist.

Das Diskettennamen-Bit kennzeichnet einen besonderen Eintrag, der für GEMDOS keine spezielle Bedeutung hat und als Diskettenname verwendet werden kann. Er ist in Directories normalerweise unsichtbar. Er wird im allgemeinen beim Formatieren festgelegt und vom Desktop beim "Disk-Info" angezeigt.

Das Ordner-Bit zeigt an, daß es sich bei der Datei um ein Directory handelt. Da Directories in einigen Punkten anders behandelt werden, sollte man es tunlichst unterlassen, mit 'Fattrib' eine "normale" Datei in einen Ordner umzuwandeln oder umgekehrt (GEMDOS fängt dies nicht ab!). Gleiches gilt auch für den Diskettennamen.

Das Archiv-Bit hat unter GEMDOS keinerlei Funktion. Normalerweise sollte es dazu dienen, veränderte Dateien zu erkennen. Backup-Programme oder ähnliches würden bei jedem Backup das Bit (bei den Quelldateien) löschen. Wenn die Datei verändert wird, müßte es gesetzt werden. Beim nächsten Backup könnte man erkennen, ob die Datei inzwischen verändert oder neu angelegt wurde.

Ein paar Worte zu den Kombinationsmöglichkeiten der einzelnen Bits sind vielleicht auch ganz angebracht. Bei Diskettennamen und Ordnern werden alle anderen Bits ignoriert, was ihre Funktion angeht. Allerdings dürfen Schreibschutz- und Archiv-Bit trotzdem nicht gesetzt sein, da Ordner sonst nicht nur als Ordner sondern auch als normale Dateien gefunden werden, was zu Konflikten führen kann.

Atari-Dokumentationen verlangen sogar, daß bei Directories und Diskettennamen keinerlei weitere Bits gesetzt werden. Bei versteckten Dateien dürfen zwar das Schreibschutz- und das Archiv-Bit gesetzt sein, doch sind dann die Dateien nicht mehr unsichtbar! System- und Versteckt-Bit dürfen miteinander kombiniert werden. Dann wird die Datei sowohl als System- als auch als versteckte Datei gefunden. Sie sehen also, daß das Dateiattribut keine allzu aufregende Sache ist.

Suchattribute

Die Directory-Suchfunktionen bekommen ein "Suchattribut" übergeben, das festlegt, unter welchen Bedingungen Dateien als gefunden gemeldet werden. Intern verwendet GEMDOS allerdings ein Suchattribut etwas anderer Form. Es wird von der Suchroutine 'd_srcname' wie folgt ausgewertet:

- Dateien mit dem Attribut 0 werden immer gefunden (unabhängig vom internen Suchattribut), außer wenn das interne Suchattribut 8 ist.
- Dateien, bei deren Attribut mindestens ein Bit gesetzt ist, das auch im internen Suchattribut gesetzt ist, werden gefunden.

Der Programmierer hat mit Suchattributen nur bei 'Fsfirst' zu tun. Das dort übergebene Suchattribut wird in das interne Suchattribut umgewandelt, indem die Bits 0 und 5 gesetzt werden, außer wenn das Suchattribut 8 ist. In diesem Fall ist das interne Suchattribut ebenfalls 8. Dies führt zu folgender Ergänzung:

- Zusätzlich zu den obigen Möglichkeiten wird eine Datei auch dann gefunden, wenn Bit 0 oder 5 seines Attributs gesetzt sind, unabhängig vom Suchattribut.

Diese Suchregeln sind allerdings nicht immer die vernünftigsten. Ein paar Beispiele für das, was so alles gefunden wird.

Suchattribut 0:
- alle "normalen" Dateien, ob mit oder ohne Schreibschutz oder Archiv-Flag
- versteckte und System-Dateien, Ordner oder Diskettenname mit Schreibschutz oder Archiv-Flag

Suchattribut $08:
- alle Arten von Diskettennamen

Suchattribut $10:
- alle Arten von "normalen" Dateien
- alle Arten von Ordnern

Suchattribut $11:
- alle Ordner und alle Dateien mit oder ohne Schreibschutz (nicht etwa alle "schreibgeschützten Ordner"!)

Suchattribut $02:
- alle normalen Dateien, versteckte Dateien
- Systemdateien, die zusätzlich noch versteckt sind
- versteckte Ordner

Ungünstig ist vor allem die Mixtur aus verschiedenen Arten von Dateien (normal, Directory, Diskettenname, Systemdatei) und ihren Eigenschaften (Schreibschutz, versteckt, archiviert). Hier gäbe es sinnvollere Regeln, die es erlauben würden, gezielt nach einzelnen Dateitypen mit bestimmten Eigenschaften zu suchen. Im allgemeinen wird man daher mit Such-

attribut 0 oder $10 arbeiten.

Interessant ist es noch zu wissen, mit welchen Suchattributen die verschiedenen Teile des TOS arbeiten. GEMDOS verwendet bei allen Dateifunktionen, die 'd_srcname' benutzen (z.B. 'Fopen' usw.), das Suchattribut $27. Damit werden sämtliche "normalen" Dateien gefunden. Es ist daher meistens möglich, lästige Zusatzdateien zu "verstecken". Das räumt so manches Directory auf. Ein gutes Beispiel ist der Auto-Boot-Harddisk-Treiber "SH204DVR.SYS", den man so elegant "loswerden" kann. Bei der Suche nach Directories benutzt GEMDOS Suchattribut $10, wie nicht anders zu erwarten.

'Frename' überprüft mit 'Fsfirst', ob der gewünschte Zielname schon existiert. Dabei wird Suchattribut 0 verwendet, was zu Problemen führt (s. 'Frename' in Teil 1).

Das AES sucht Accessories mit 'Fsfirst' und Suchattribut 0. "Versteckte" Accessories werden daher nicht gefunden.

Das DESKTOP.INF dagegen wird mit 'Fopen' angesprochen und kann daher "versteckt" werden (und wieder wird ein Directory etwas übersichtlicher).

Beim Suchen von Resource Dateien wird Suchattribut 5 genommen (findet auch "System"-Dateien). Also können auch die oft lästigen Resources unsichtbar gemacht werden.

Beim Arbeiten mit Command Line-Interpretern gibt es manchmal eine Option, wirklich alle (also auch die versteckten) Dateien aufzulisten. Der Desktop kann dies nicht, daher ist es zu empfehlen, das Verstecken auf wenige Directories zu beschränken (z.B. auf die Root-Directories der Harddisk), damit man nicht den Überblick verliert.

Ermitteln eines Directories mit d_getdir

Kommen wir nun zur Routine 'd_getdir', die von den meisten GEMDOS-TRAP1-Funktionen benutzt wird. Sie ermittelt aus einem Dateipfad den DD des zugehörigen Directories und eventuell den aus dem Pfad isolierten eigentlichen Dateinamen.

Der grobe Ablauf ist wie folgt: Der Pfadname wird in einzelne Directory-Namen zerlegt, wobei sich GEMDOS im DD-Baum von den oberen zu den unteren Ebenen hangelt. Doch nun zu den Details.

Zuerst wird das Start-Directory bestimmt, auf das sich der restliche Pfad bezieht. Dazu wird als allererstes das Laufwerk festgelegt. Wenn es im Dateipfad nicht explizit angegeben ist, wird das aktuelle Laufwerk ('p_defdrv' im PD) ausgewählt. Wie im Teil 1 beschrieben, übernimmt 'd_chkdrv' die Überprüfung des Laufwerks und sorgt für die Existenz eines Standardpfades. Beginnt der Pfad mit einem '\', so ist das Root-Directory das Start-Directory, ansonsten das Ende des Standardpfads. Das so festgelegte Start-Directory ist ersteinmal das "aktuelle Directory".

Nun wird die nächste Komponente des Pfades extrahiert und in das GEMDOS-interne Format konvertiert. Die Spezial-Dateinamen '.' und '..' werden hier separat ausgewertet. '.' wird ignoriert, bei '..' wird das Parent-Directory des aktuellen Directories über den 'dd_pardd'-Zeiger ermittelt. Unschönerweise erfolgt hier keine Sicherheitsabfrage ob man sich nicht schon im Root-Directory befindet. Da dessen 'dd_pardd' NIL ist, gibt es gleich darauf einen Bus-Error.

Ansonsten muß das gewünschte Subdirectory im aktuellen Directory gesucht werden. Wenn GEMDOS in seinem "DD-Baum" (s. Teil 1) hier noch überhaupt kein Child kennt, muß auf jeden Fall im Directory selbst gesucht werden. Dies erledigt 'd_srcname' (s.u.). Wenn im aktuellen Directory kein weiteres Directory ist, ist die Suche auf jeden Fall fehlgeschlagen und wird abgebrochen.

Wenn schon mindestens ein Child bekannt ist oder von 'd_srcname' eins gefunden wurde, wird die Child-Liste des DD durchsucht. Falls das gewünschte Directory hier nicht gefunden werden kann, besteht noch die Möglichkeit, daß noch nicht alle Child-Directories in der Child-Liste eingetragen sind. Dies wird am 'fd_dirch'-Flag des Directory-FD erkannt (s.u.). In diesem Fall wird nochmals 'd_srcname' aufgerufen. Schlägt dies auch fehl, so existiert das Directory nicht und es wird abgebrochen.

Nun wird das ermittelte Child zum aktuellen Directory und das Spiel beginnt von vorne, bis der ganze Pfad durchgearbeitet ist.

Je nach Aufruf (bestimmt durch ein Flag) wird dabei der ganze String ('Dsetpath') oder alles bis auf den letzten Namen ('Fopen') als Pfadname angesehen. Im zweiten Fall wird der letzte Name als Dateiname interpretiert und der übergeordneten Routine zur Verfügung gestellt.

Letztendlich erhält man also den DD des gewünschten Directories am Pfadende, wobei alle Directories auf dem Weg dorthin automatisch im DD-Baum angelegt werden.

Suchen von Dateien mit d_srcname

Doch nun zur internen Funktion 'd_srcname', die die Drecksarbeit, sprich das Suchen im Directory, erledigt und so nebenbei den DD-Baum aufbaut. 'd_srcname' wird für alle Sucharbeiten (Dateinamen, Directories oder freie Einträge) verwendet und bekommt daher auch das "interne Suchattribut" übergeben.

Da hier mit den üblichen Dateioperationen auf das Directory zugegriffen wird, muß ersteinmal ein FD für das Directory erzeugt werden, falls noch keiner vorhanden ist.

'd_srcname' kann eine Anfangs-Dateiposition übergeben werden, ab der gesucht werden soll. Dies ist wichtig, wenn erst ab einer bestimmten Stelle gesucht werden soll (wie bei 'Fsnext').

'd_getdir' läßt dagegen immer ab der letzten Suchposition ('dd_lpos') suchen, da hier ja nur die Child-Liste verlängert werden soll, und 'dd_lpos' gerade angibt, bis wohin die Subdirectories schon erfaßt wurden.

Das Directory wird nun ab dieser Anfangs-position Eintrag für Eintrag gelesen, bis das Ende erreicht wird oder bis die gesuchte Datei gefunden wurde. Das Ende eines Directories wird von GEMDOS nicht nur an seiner tatsächlichen Länge in Sektoren bzw. Clustern erkannt. Schon der erste Eintrag, der mit einem Nullbyte beginnt, führt bei Directory-Suchoperationen zum Abbruch. Dies spart vor allem beim Root-Directory erheblich an Zeit.

Zuerst wird überprüft, ob der gerade untersuchte Eintrag ein geeigneter Kandidat für die automatische Aufnahme in den DD-Baum ist. Dazu muß es sich natürlich um einen Directory-Namen (außer '.' und '..') handeln. Außerdem muß die aktuelle Dateiposition hinter 'dd_lpos' liegen, da das Directory sonst schon im Baum vorhanden ist. Desweiteren wird überprüft, ob nicht überhaupt schon früher einmal das gesamte Directory durchsucht wurde ('fd_dirch'-Flag gesetzt, s.u.). Wenn nach einem Directory (nicht nach einer normalen Datei) gesucht wird, soll eine letzte Abfrage nochmals sicherstellen,

daß der Ordner nicht schon in der Child-Liste vorhanden ist.

Nachdem alle Klippen überwunden sind und 'd_srcname' der Meinung ist, das Directory müsse in die Child-Liste des durchsuchten Directories aufgenommen werden, wird ein DD eingerichtet. Die benötigten Daten werden dem Directory-Eintrag bzw. dem DD des Parents entnommen. Der neue DD wird immer vorne in die Child-Liste eingehängt.

Nun erfolgt der Vergleich des Directory-Eintrags mit den Suchkriterien. Eine Datei gilt als gefunden, wenn ihr Name unter Berücksichtigung der Ersetzungszeichen und ihr Attribut im Rahmen der Suchregeln übereinstimmt.

Bei der Suche nach einem freien Eintrag (z.B. von 'Fcreate' aus) wird 'd_srcname' fündig, wenn es auf $00 oder $E5 im ersten Byte des Dateinamens stößt.

Nach Abbruch der Suche (egal ob erfolgreich oder nicht) wird 'dd_lpos' auf den neuesten Stand gebracht, aber nur, wenn 'd_srcname' von 'd_getdir' aus aufgerufen wurde. Dies führt zu einem fatalen Fehler, wie unten noch beschrieben wird.

Wenn die gewünschte Datei nicht gefunden werden konnte und auch nicht gerade nach einem freien Eintrag gesucht wurde, so wird das Bit 1 von 'fd_dirch' im FD des Suchdirectories gesetzt. Es zeigt somit an, daß das Directory einmal komplett gelesen wurde und dieser Zweig des DD-Baums vollständig aufgebaut ist.

Je nach Aufruf von 'd_srcname' wird ein Zeiger auf den Directory-Eintrag der gefundenen Datei oder einer auf den DD des gefundenen oder in die DD-Liste aufgenommenen Directories (NIL zeigt Fehlschlag an) zurückgegeben.

Fehler in der Directory-Verwaltung

Diese Routine beinhaltet einen der schwerwiegendsten Fehler des ganzen GEMDOS. Er kann dazu führen, daß ein Directory *mehrmals* in die Child-Liste seines Parents aufgenommen wird.

Zum einen führt dies dazu, daß zuviel "interner Speicher" für die DDs und eventuell deren FDs verbraucht wird. Dadurch findet eine zusätzliche Begrenzung der ansprechbaren Ordner statt (das berühmt-berüchtigte "40-Ordner-Problem"). In vielen Fällen tritt dieses Problem überhaupt nur auf Grund dieses Fehlers auf und nicht, weil zu viele Ordner geöffnet wurden o.ä!

Es sind aber noch weitere merkwürdige Effekte möglich. Es kann passieren, daß sich ein Directory nicht mehr löschen läßt, obwohl es keine Dateien mehr enthält, denn beim 'Ddelete' wird überprüft, ob die Child-Liste des zu löschenden Directories auch wirklich leer ist (s. Teil 1).

Doch nun zur Ursache des Fehlers. Beim Aufruf von 'd_srcname' für "normale" Suchoperationen (wie etwa von 'Fopen', 'Fcreate' usw.) werden ebenfalls "zufällig" entdeckte Directories in den DD-Baum integriert. Im Gegensatz zu der von 'd_getdir' initiierten Suche wird aber 'dd_lpos' nicht auf die Endposition der Suche gesetzt, so daß GEMDOS bei der nächsten Suchoperation nicht merkt, daß das Directory schon in der Child-Liste hängt und es erneut einfügt.

Warum, so werden Sie sich vielleicht fragen, merkt man dann normalerweise nicht viel von diesem Fehler? Er müßte sich doch eigentlich ständig bemerkbar machen.

Dies liegt daran, daß die Child-Liste nicht mehr erweitert wird, wenn das 'fd_dirch'-Bit 1 gesetzt ist. Das ist aber der Fall, sobald das Directory einmal komplett durchsucht wurde, weil eine Suchoperation fehlgeschlagen ist. Das ist aber genau das, was beim Arbeiten mit dem Desktop ständig passiert. Meistens macht man einen Ordner auf, bevor man mit ihm arbeitet. Dabei liest der Desktop das Directory komplett, um es im Fenster darzustellen.

Auch bei Command Line-Interpretern listet man die meisten Directories irgendwann einmal, oft ebenfalls bevor man sie anspricht.

Gefährlich wird es bei Programmen, die in einem Directory ständig ganz bestimmte Dateien bearbeiten. Da dabei i.allg. keine Suchoperation fehlschlägt, weil das Programm ja "gezielt" sucht, geht bei jedem Datei-Öffnen interner Speicher verloren. Da dieser Effekt keine Begrenzung kennt, kommt es irgendwann zum Zusammenbruch der internen Speicherverwaltung, da hilft dann kein noch so großzügiges "Mehr-Ordner-Programm".

Ein schönes Beispiel findet sich in Listing 1. Vor Ablauf des Programms sind allerdings einige Vorbereitungen notwendig. Legen Sie in dem Directory, in dem sich das Programm befindet, einen Ordner "TEST.ORD" an. Öffnen Sie ihn nun und erzeugen Sie zuerst einen weiteren Ordner mit beliebigem Namen. Außerdem muß danach irgendeine Datei in den Ordner "TEST.ORD" kopiert und in "TEST" umbenannt werden. Initialisieren Sie das System mittels RESET-Knopf neu und starten Sie das Demo-Programm, ohne vorher "TEST.ORD" geöffnet zu haben. Nach einiger Zeit meldet das Programm einen Fehler und gibt an, wie oft es die Datei "TEST" öffnen und wieder schließen konnte. Dieser Wert hängt vom verwendeten TOS ("altes TOS" oder "Blitter-TOS") und eventuell vorhandenen "Mehr-Ordner-Programmen" ab.

Nun ist der GEMDOS-Speicher fast restlos aufgebraucht, und Sie werden dies am merkwürdigen Verhalten des Desktops bemerken. Ordner sind scheinbar leer, es gibt ungewohnte Fehlermeldungen bei harmlosen Operationen, Abstürze oder einen selbstausgelösten RESET.

Warnung: Führen Sie keinesfalls irgendwelche Schreibzugriffe durch (wie Kopieren), da sonst die Gefahr von ungewollten Datenverlusten besteht.

Am besten machen Sie einen RESET, dann normalisiert sich wieder alles. Wenn "TEST.ORD" vor dem Start des Demos einmal geöffnet wurde, geht alles gut und das Programm bricht ohne Fehler nach 1000 'Fopen'-'Fclose'-Sequenzen ab.

Die "Disk Transfer Area" (DTA)

GEMDOS stellt dem Programmierer zwei Funktionen zur Verfügung, mit denen ein Directory nach bestimmten Einträgen durchsucht werden kann. Im einfachsten Fall dienen sie dem Einlesen des kompletten Directories, was z.B. der Desktop macht, um es dann in einem Fenster darzustellen.

'Fsfirst' führt dabei die erste Suche aus, die am Anfang des Directories beginnt, und hinterlegt die im Directory gespeicherten Informationen über die gefundene Datei in der DTA.

'Fsnext' ist in der Lage, ab der Stelle weiterzusuchen, wo 'Fsfirst' bzw. ein vorheriges 'Fsnext' aufgehört hat. Durch wiederholte Aufrufe ist es damit möglich, ein Directory komplett nach Dateien, die bestimmten Kriterien genügen, zu durchsuchen.

Die DTA enthält daher auch alle Daten, die GEMDOS braucht, um die Suche am alten Abbruch-Punkt aufsetzen zu lassen. Speicherplatz für die DTA kann vom Anwender selbst bereitgestellt werden (s. 'Fsetdta' und 'Fgetdta'). Eine genaue Beschreibung dieser Funktionen findet

```
typedef struct
{ char dta_sname[12];          /* Suchname (von Fsfirst) */
  char dta_sattr;              /* Suchattribut (von Fsfirst) */
  char dta_dpos[4];            /* letzte Suchposition im Suchdirectory */
  char dta_dd[4];              /* Zeiger auf DD des Suchdirectories */
  char dta_attr;               /* gefundenes Attribut */
  unsigned int dta_time;       /* gefundene Zeit */
  unsigned int dta_date;       /* gefundenes Datum */
  long dta_len;                /* gefundene Laenge */
  char dta_name[14];           /* gefundener Name */
} DTA;
```

Abb. 1: Struktur der "Disk Transfer Area" (DTA)

sich im Abschnitt "GEMDOS-Funktionen".

Hier geht es in erster Linie um den Aufbau der DTA (Abb. 1). Beginnen wir hinten, da dort die für den Programmierer wichtigen Daten abgelegt sind.

'dta_name' ist der Name der gefundenen Datei in der dem Anwender bekannten Darstellung (nicht im internen Format). 'dta_name' ist mit 14 Zeichen großzügig dimensioniert (13 hätten auch gelangt). Desweiteren findet man hier die Dateilänge in Bytes ('dta_len'), Erstellungszeit ('dta_time') und -datum ('dta_date') sowie das Dateiattribut ('dta_attr') der gefundenen Datei. Diese Komponenten sind alle von Atari dokumentiert, anders wäre die Nutzung von 'Fsfirst'/'Fsnext' wohl auch kaum möglich.

Der Rest der DTA ist offiziell als "reserviert für interne Zwecke" ausgewiesen, daher kann (und nach neuesten Informationen soll) sich die Belegung in späteren TOS-Versionen durchaus ändern. Man sollte daher die Kenntnis hiervon nicht beim Programmieren ausnutzen.

Der bei 'Fsfirst' übergebene Suchname (abgetrennt vom zugehörigen Suchpfad), wird in 'dta_sname' abgelegt. Dieser Name liegt im GEMDOS-internen Format vor. Das bei 'Fsfirst' festgelegte Suchattribut wird in 'dta_attr' aufbewahrt. 'dta_dd' zeigt auf den DD des Directories, in dem die Suche stattfindet.

Die aktuelle Dateiposition der Suche ist in 'dta_dpos' zu finden. 'dta_dpos' zeigt dabei auf den Eintrag nach dem zuletzt gefundenen, also auf den, bei dem die nächste Suche beginnen soll.

Bei 'dta_dd' und 'dta_dpos' fällt die merkwürdige Definition als "char-Array" in der Struktur auf. Etwas besseres fiel mir nicht ein, da es sich offensichtlich um 32-Bit-Werte auf ungeraden Adressen handelt. So etwas erzeugt kein C-Compiler, daher muß die Original-Definition von Digital Research wohl ähnlich merkwürdig aussehen. Der Zugriff auf diesen Teil der DTA geschieht ohnehin nur mit Byte-Kopierroutinen.

Das Problem wäre bei anderer Anordnung der Komponenten oder Verkürzung von 'dta_sname' (11 Zeichen reichen auch) nicht aufgetreten.

Mit Hilfe von 'dta_dd' kommt man übrigens an die internen Datenstrukturen der Dateiverwaltung heran. Hier findet sich nämlich unter anderem ein Zeiger auf den DMD, von dem aus alle DDs und FDs erreichbar sind. Dies kann als eine "halblegale" Methode angesehen werden, da zwar nicht die Kenntnis von veränderlichen Systemadressen, aber die der internen Strukturen vorausgesetzt wird.

Der DMD ist auch über die Sektorpuffer-Listen ('bufl') erreichbar, doch kann man oft nicht wissen, ob ein solcher Puffer für das gewünschte Laufwerk existiert.

GEMDOS-Funktionen (TRAP #1)

Hier sind die vier GEMDOS-Funktionen zusammengestellt, die mit Suchoperationen zu tun haben.

Funktion $1A Fsetdta

*void Fsetdta(DTA *new_dta)*

Die Adresse der aktiven DTA wird auf 'new_dta' gesetzt. Die DTA wird von 'Fsfirst'/'Fsnext' benutzt. Vor dem Start jedes Prozesses wird eine Default-DTA definiert. Sie liegt bei der aktuellen GEMDOS-Version im Prozeßdescriptor unmittelbar am Beginn der Kommandozeile.

In vielen Fällen muß man daher keinen eigenen Speicherbereich mit 'Fsetdta' zur Verfügung stellen. Man muß aber Konflikte mit der Kommandozeile berücksichtigen. Das bedeutet, die Kommandozeile muß vor dem ersten 'Fsfirst' ausgewertet sein, wenn keine eigene DTA verwendet wird.

Arbeitsweise

Hierbei handelt es sich um einen 'C'-Einzeiler. 'new_dta' wird direkt in den PD des aktiven Prozesses nach 'p_dta' übertragen.

Funktion $2F Fgetdta

*DTA *Fgetdta()*

Die Adresse der aktiven DTA wird zurückgegeben. Es ist immer eine DTA definiert (s. 'Fsetdta').

Arbeitsweise

Noch ein Einzeiler: Der Rückgabewert ist einfach 'p_dta' des aktiven Prozesses.

Funktion $4E Fsfirst

*long Fsfirst(char *path, int attr)*

Das durch den Directory-Anteil von 'path' ausgewählte Directory wird nach Dateien durchsucht, deren Namen durch den von 'path' abgetrennten Suchnamen und das Suchattribut 'attr' bestimmt werden.

Beispiel: Bei einem 'path' von 'ORDNER*.DOC' wird im Directory 'ORDNER', das sich im aktuellen Directory des Standardlaufwerks befindet, nach allen Dateien gesucht, die die Endung '.DOC' haben, und mit dem Suchattribut vereinbar sind.

Die Suche wird beim Anfang des Directories begonnen und bricht beim ersten die Suchanfrage erfüllenden Dateinamen ab. Sie kann mit 'Fsnext' fortgeführt werden. Der Suchname kann die üblichen Ersetzungszeichen '?' und '*' enthalten, insbesondere sucht '*.*' nach allem. Das Suchattribut ist oben schon erklärt worden.

Rückgabewerte:

-1..-31	BIOS-Fehlermeldung bei Diskzugriff
-33L	(EFILNF) Pfad oder keine Datei gefunden, Laufwerk existiert nicht, interner Fehler (keine Pfad-Handles, zu wenig interner Speicher)
0L	alles ok.

Arbeitsweise

Das Suchattribut wird in das interne Suchattribut konvertiert, wie oben beschrieben.

'd_getdir' ermittelt das Suchdirectory und mit 'd_srcname' wird ab dem Directory-Anfang wie gewünscht gesucht. Wenn eine Datei gefundenen wurde, die die Suchanfrage erfüllt, wird die DTA

komplett gefüllt (Teile für Anwender und interne Zwecke).

Funktion $4F Fsnext

long Fsnext()

Eine vorher mit 'Fsfirst' begonnene Suche wird fortgesetzt. Die Suche wird ab dem Eintrag fortgeführt, der dem letzten gefundenen folgt. 'Fsnext' ist mehrmals hintereinander anwendbar, bis das ganze Directory durchsucht ist.

Andere Dateioperationen "zwischendurch" sind durchaus erlaubt. Fatal ist dagegen ein Mediumwechsel. Er sorgt für die Freigabe aller internen Strukturen. Bei 'Fsnext' wird auf diese zugegriffen, obwohl sie nicht mehr definiert sind. Dies kann zur Suche in einem ganz anderen Directory oder zum Absturz führen. Ebenfalls tödlich ist ein 'Fsnext' ohne vorhergehendes 'Fsfirst', da die DTA dann undefiniert ist.

Rückgabewerte:

-1..-31	BIOS-Fehlermeldung bei Diskzugriff
-49L	(ENMFIL) keine Datei gefunden, interner Fehler (zu wenig interner Speicher)
0L	alles ok.

Arbeitsweise

Mit den in der DTA gespeicherten Daten wird 'd_srcname' aufgerufen. Wenn die Suche erfolgreich war, wird der Anwender-Teil der DTA und 'dta_dpos' neu gesetzt.

Anwendungsbespiel

Ein Beispiel für die Verwendung dieser vier GEMDOS-Funktionen ist in Listing 2 gegeben.

Die Funktion 'search' sucht in einem bestimmten Directory nach sämtlichen Dateien (außer dem Diskettennamen). Dabei werden auch alle Subdirectories sowie deren Subdirectories usw. durchsucht. Bei jeder gefundenen Datei wird 'found' aufgerufen, das in diesem Beispiel nichts weiter macht, als den Pfad und das Dateiattribut der gewünschten Datei auszugeben. Aufgerufen wird 'search' von 'main', wobei das aktuelle Directory als zu durchsuchendes festgelegt wird. Die Einzelheiten können dem Listing entnommen werden.

Mit Vorsicht ist dieses Programm im Root-Directory einer Harddisk anzuwenden. Da dabei sämtliche Ordner geöffnet werden, tritt ein erheblicher Verbrauch an "internem Speicher" auf, womit wir wieder beim "40-Ordner-Problem" angelangt wären.

Ausblick

Das Gröbste hätten wir nun hinter uns. Zur Datei- und Directory-Verwaltung wissen Sie nun alles wesentliche. Was noch bleibt, sind vor allem die I/O-Umleitung und die I/O-Funktionen für die zeichenorientierten Geräte, womit wir uns als nächstes beschäftigen werden.

Alex Esser

Listings folgen nächste Seite

```
 1: /* Demo-Programm f.TOS-Fehler d.Directory-Verwaltung */
 2:
 3: #include <osbind.h>
 4:
 5: main()
 6: {  int err;
 7:    int i;
 8:
 9: /* maximal 1000 Durchläufe oder Abbruch durch Fehler */
10:    for (i=0,err=0; err==0 && i<1000; i++)
11:    {
12:      err = Fopen("test.ord\\test",0);
13:       if (err > 0)
14:          err = Fclose(err);
15:    }
16:    printf("Abbruch nach %d Durchläufen mit Fehler %d.\n"
                                               ,i,err);
17:    Cnecin();
18: }
```

Listing 1: Demo-Programm für TOS-Fehler

```
 1: /* Rekursives Durchsuchen eines Directories
 2:     by A Esser
 3:     13 6 1988
 4:  */
 5:
 6: #include <osbind h>
 7:
 8: #define E_OK         0
 9: #define ERROR        1
10: #define MAX_RECUR    10      /* max  Rekursionstiefe */
11:
12:
13: typedef struct
14: {  char dta_reserved[21];
15:    char dta_attr;
16:    unsigned int dta_time;
17:    unsigned int dta_date;
18:    long dta_len;
19:    char dta_name[14];
20: } DTA;
21:
22:
23: /* Rekursiv durch Directory-Hierarchie durcharbeiten */
24: /* 'path' ist Start-Pfad (ohne abschließendes '\') */
25: /* Erst-Aufruf erfolgt mit 'level'=0 */
26: int search(level,path)
27: int level;        /* Rekursionstiefe */
28: char *path;
29: {  DTA dta;       /* lokale DTA */
30:    DTA *olddta;  /* DTA-Adresse der höheren Ebene */
31:     int err;
32:     char *name;   /* Zeiger auf Dateinamen in 'path' */
33:
34:    olddta = (DTA *)Fgetdta(); /* alte DTA-Adresse
                                          merken */
35:    Fsetdta(&dta);  /* lokale DTA definieren */
36:     name = path + strlen(path) + 1;
37:     strcpy(name-1,"\\*.*");            /* alles suchen */
38:    err = Fsfirst(path,0x37);
39:    while (err == E_OK) /* suchen bis nichts mehr da
                                     oder Fehler */
40:     {
41:       strcpy(name,dta.dta_name); /* gefundenen Namen
                                       an Pfad anhängen */
42:        found(path,&dta);          /* gewünschte Aktion
                                        ausführen */
43:        if (dta.dta_attr & 0x10  &&  *name != '.')
44:           if (level < MAX_RECUR)  /* ab in die nächste
                                          Ebene */
45:              err = search(level+1,path);
46:           else
47:              err = ERROR; /* Rekursion zu tief Abbruch */
48:        if (err == E_OK)
49:           err = Fsnext(); /* weiter suchen wenn kein
                                    Abbruch */
50:     }
51:    Fsetdta(olddta); /* DTA der letzten Ebene zurück */
52:     if (err < 0)     /* keine Datei mehr gefunden */
53:        err = 0;      /* weiter in letzter Ebene */
54:     return err;      /* sonst Abbruch */
55: }
56:
57:
58: /* Aufruf bei jeder gefundenen Datei einschl. Ordner */
59: /* Beispiel: Ausgabe des Pfades und des Attributs */
60: found(path,dta)
61: char *path;
62: DTA *dta;
63: {
64:    printf("$%02x  %s\n",dta->dta_attr,path);
65: }
66:
67:
68: /* Beispiel für Aufruf */
69: main()
70: {
71:    char cpath[128];
72:
73:    Dgetpath(cpath,0); /* los geht's ab akt. Directory */
74:    search(0,cpath);
75: }
```

Listing 2: Anwendung der Datei-Suchfunktionen

ENDE

1ST_WORDPLUS NACH TeX

Das Satzprogramm TeX ist herkömmlichen Textverarbeitungen wie WordPlus in vielen Aspekten überlegen, doch wer umsteigt, muß seine Texte neu erfassen. Abhilfe schafft das hier abgedruckte Programm, das alle Wordplus-Stilarten und -Zeichen in TeX-Kommandos umsetzt.

Nach dem Programmstart geben Sie den Namen der zu konvertierenden Datei ein. TeXConverter erzeugt daraus eine ASCII-Datei unter gleichem Namen mit der Endung .TEX, die als Eingabe für TeX verwendet werden kann. Während der Umsetzung zeigt TeXConverter mit der Ausgabe kleiner Sternchen den Fortgang der Arbeit an (Bild 1).

Als Eingabe können prinzipiell alle WordPlus-Dokumente verwendet werden. Das Programm verarbeitet alle Textarten und Symbole des ST-Zeichensatzes. Die Ausgabe enthält die entsprechenden TeX-Kommandos und ist mit der Einschränkung, daß das £-Zeichen anders angesprochen werden, muß auch für LaTeX geeignet.

Auf jeden Fall aber sind Feinheiten nach einem TeX- und DVI-Lauf nochmals zu überarbeiten, denn der gewiefte TeXniker ist in typographischer Hinsicht bekanntlich ein Perfektionist. Kleinere Dinge, wie die Italic-Correction werden aber automatisch erzeugt.

```
 !"#$%&'()*+,-./0123456789:;<=>?
@ABCDEFGHIJKLMNOPQRSTUVWXYZ[\]^_
`abcdefghijklmnopqrstuvwxyz{|}~Δ
ÇüéâäàåçêëèïîìÄÅÉæÆôöòûùÿÖÜ¢£¥ßƒ
áíóúñÑªº¿⌐¬½¼¡«»ãõØøœŒÀÃÕ¨´†¶©®™
ĳĲאבגדהוזחטיכלמנסעפצקרשתןךםףץ§∧∞
αβΓπΣσµτΦΘΩδ∮φ∈∩≡±≥≤⌠⌡÷≈°•·√ⁿ²³
```

Bild 2a: Vorher - der komplette Zeichensatz in 1st_WordPlus

```
!"#$%&'()*+,-./0123456789:;<=>?
@ABCDEFGHIJKLMNOPQRSTUVWXYZ[\]^_
'abcdefghijklmnopqrstuvwxyz{|}~Δ
ÇüéâäàåçêëèïîìÄÅÉæÆôöòûùÿÖÜ ¢£¥ßƒ
áíóúñÑªº¿⌈⌉ 1/2 1/4 '≪≫ã õØøœŒÀÃÕ¨ ´ †¶©®™
ĳĲℵ********* *********** ******§∧∞
αβΓπΣσµτΦΘΩ δ*φ∈∩≡±≥≤**÷ ≈∘•.√n23
```

Bild 2b: Nachher - derselbe Zeichensatz in TeX

Einschränkungen

In Bild 2 sehen Sie die Ausgabe aller Zeichen, die mit WordPlus erzeugt werden, können, und den entsprechenden TeX-Ausdruck. Nicht verfügbare Symbole sind mit einem Stern (\star) gekennzeichnet. Dies betrifft allerdings hauptsächlich die hebräischen Zeichen, an deren Stelle aber auch andere eingefügt werden können.

Die Verarbeitung mit Plain-TeX ist problemlos. Wenn Sie LaTeX benutzen, müssen Sie das Listing entsprechend den Kommentaren anpassen. Alle Voreinstellungen sind im Programm durch Konstanten veränderbar. Dazu müssen Sie natürlich ein Modula-2 TeXniker sein...

Fußnoten, die mit WordPlus erzeugt wurden, werden nicht übersetzt, da dazu ein zweiter Durchlauf über die Eingabe nötig würde. Die Markierungen werden einfach überlesen; die eigentliche Fußnote erscheint im laufenden Text und muß mit einem Editor in ein \footnote-Kommando bewegt werden.

```
WordPlus -> TeX Converter
  1988 Robert Tolksdorf

WordPlus-Filename : texconv.doc

texconv.doc --> texconv.TEX

*******
```

Bild 1: Der Converter bei der Arbeit

geht weiter...

TeXniken

TeXniker und TeXacker sind Perfektionisten. Vielleicht sind einige Konvertierungen noch zu verbessern, darum hier ein paar Anmerkungen zum Programm.

Der Converter (Listing 1) versucht, überflüssige Ausgaben zu vermeiden (überhaupt ist einige Intelligenz eingebaut). Mehrfache Blanks sind für TeX unsinnig; das Flag lastwasblank sorgt dafür, daß doppelte Leerzeichen, die durch den Randausgleich in WordPlus entstehen, überlesen werden. Analog dazu wird über lastwaspar vermieden, daß mehrfache Leerzeilen zur Absatztrennung in die Ausgabe kommen.

Die Schriftarten in WordPlus werden entsprechend übersetzt. Dabei erscheint Kursiv als \it und die Hell-Schrift als \tt. Die "Italic-Correction" \/ wird automatisch eingefügt. Für Indizes und Potenzen wird in den Math-Mode geschaltet, die Schriftart wird dabei auf \rm gesetzt.

In WordPlus können mehrere Schriftauszeichnungen miteinander kombiniert werden. Da bei TeX z.B. Fettschrift nicht errechnet wird, sondern - typographisch richtiger - in einem eigenen Zeichensatz bereitliegt, ist eine Vermischung von Auszeichnungen nicht möglich. Der TeXConverter wählt nach der Reihenfolge Fett-Hell-Kursiv-Unterstrichen-Hochgestellt-Tiefgestellt eine Schriftart aus, falls in WordPlus mehrere gleichzeitig eingestellt wurden.

TeX kann als eine der wenigen Textverarbeitungen zwischen öffnenden und schließenden Anführungszeichen unterscheiden. Über quote wird zwischen den entsprechenden Kommandos " und " umgeschaltet. Anführungszeichen müssen also in der Eingabe immer paarweise vorkommen - ein einzelnes bringt den restlichen Text etwas durcheinander.

Alle Zeichen des Atari-Satzes werden über das Feld Charmap in TeX-Kommandos umgewandelt. Symbole, die nicht übernommen werden können, erzeugt das Programm nach der Konstante NA, die auf \star voreingestellt ist. Die nicht erzeugbaren Zeichen sind entweder für TeX unsinnig, wie z.B. die "halben" Integralzeichen, oder nicht möglich, so die hebräische Schrift. Bei entsprechender Veränderung der Tabellen können Sie benötigte Sonderzeichen unter den noch freien Einträgen verstecken.

Bei einigen Sonderzeichen des ST mußten kleine Tricks angewandt werden. Das Cent-Zeichen entsteht durch Überlagerung von c und / mit dem \rlap-Kommando. Auf gleiche Art wird ¥ erzeugt, nur daß hierbei Y und = übereinander gesetzt werden.

Das (nicht ganz perfekte) Zeichen für "Registriertes Warenzeichen" ist eine Abwandlung des \copyright-Befehls, bei dem anstelle des c ein R im Typewriter-Font in den Kreis kommt. Beim Wurzel-Symbol bezieht sich \sqrt auf ein Leerzeichen (\ ) und die Zeichen ^{n}, ^{2} und ^{3} sind Potenzen im Math-Modus.

Wer ein TeXmaster ist, kann die Zeichen sicherlich noch besser darstellen. In der Übersetzungstabelle lassen sich die Definitionen leicht ändern. Ich würde mich

```
msoff
echo '033E'
echo 'Willkommen beim TeX-Gulam'
uekb 07f 1d
setenv PATH gulams
rehash
set prompt 'TeX <$CWD> :'
alias dvi 'gem dvi.prg'
alias tex 'tex.ttp &plaing '
alias lat 'tex.ttp &lplaing '
alias ed 'cd inputs; proed.ttp $1.tex; cd ..'
alias ls 'ls -l'
echo ' '
echo ' Zusätzliche Befehle: '
echo ' '
echo '>- ed <fn>     : edit files <fn>.tex'
echo '>- tex <fn>    : TeX <fn>.tex mit plaing-Format'
echo '>- lat <fn>    : TeX <fn>.tex mit lplaing-Format (LaTeX)'
echo '>- dv [1|2|3|4] <fn> : printout <fn>.dvi'
echo '>-                      dv 1 <fn>  : 180*180 Ausdruck'
echo '                        dv 2 <fn>  : 360*360 Ausdruck'
echo '                        dv 3 <fn>  : 360*360 Ausdruck mit
echo '                                     separate even/odd'
echo '                        dv 4 <fn>  : 300*300 Laserausdruck
echo '                        dv <fn>    : 180*180 Ausdruck'
echo '>- dvi              : DVI als GEM-Programm'
echo ' '
mem
```

Bild 3: GULAM.G-Batchfile für Gülam-Shell

```
if { $1 == 1 }
  set dviinf 'dvidraft.inf'
  set fn $2
ef { $1 == 2 }
  set dviinf 'dviprint.inf'
  set fn $2
ef { $1 == 3 }
  set dviinf 'dviprinq.inf'
  set fn $2
ef { $1 == 4 }
  set dviinf 'dvilaser.inf'
  set fn $2
ef
  set dviinf 'dvidraft.inf'
  set fn $1
endif
ren 'dvi.inf' 'dvitemp.inf'
ren $dviinf 'dvi.inf'
time dvi.prg dvi\$fn.dvi '$batch'
ren 'dvi.inf' $dviinf
ren 'dvitemp.inf' 'dvi.inf'
unset fn
unset dviinf
```

Bild 4: DV.G-Batchfile zum vereinfachten DVI-Ausdruck

```
 1:  MODULE Converter;
 2:  (*$V+,$R+,$A+*)
 3:  (* Robert Tolksdorf, 28.01.88 - 16.4.88 *)
 4:  FROM DiskOps          IMPORT OpenFiles, CloseFiles, EOst, Get, Put ;
 5:  FROM ASCII            IMPORT CR, VT, US, RS, LF, GS, FS, FF, CAN, EM, ESC;
 6:  FROM Strings          IMPORT InitStringModule, String, Length, Concat,
 7:                                Delete ;
 8:  FROM GEMDOS           IMPORT ConOut, ConWS, ConRS ;
 9:
10:  CONST
11:    fett = 0;        (* Attribute beim Wordplus-Stil *)
12:    hell = 1;
13:    kursiv = 2;
14:    unters = 3;
15:    hochge = 4;
16:    tiefge = 5;
17:   MaxLineLength = 60;      (* Zeile umbrechen bei MaxLineLength *)
18:   NA = '$\star$';         (* Ersatz für nicht vorhandene Zeichen *)
19:
20:  TYPE SetOfChar = SET OF CHAR ;
21:      WordTypes = (Normal, Blank, StyleChange, NewPage, NewPar);
22:      TeXStyleTypes= (roman, bold, italic, underl, super, sub, typew);
23:       AWordType = RECORD CASE Type:WordTypes OF
24:                             Normal      : TheWord:String;
25:                            | Blank       :
26:                           | StyleChange : Style:TeXStyleTypes;
27:                            | NewPage     :
28:                            | NewPar      :
29:                            END;
30:                     END;
31:
32:  VAR TeXStyleBegin, TeXStyleEnd      : ARRAY [roman..typew] OF String;
33:      CharMap                         : ARRAY [32..255] OF String;
34:      Quotes                          : ARRAY [FALSE..TRUE] OF String;
35:      aktTeXStyle                     : TeXStyleTypes;
36:     TeXLine, SoftBlank, HardBlank,
37:     BeginCommand, EndCommand,
38:     NewPageCommand, Hyphenation      : String;
39:      aktchar                         : CHAR;
40:     lastwasblank,lastwasdash, quote  : BOOLEAN;
41:
42: PROCEDURE Init():BOOLEAN; (* Kommandostrings initialisieren *)
43:  VAR c:INTEGER;
44:     RegisteredMacro:String;
45:     toconvert: RECORD CASE :BOOLEAN OF  (* wg. TDI-Libraries *)
46:                        TRUE : a: ARRAY[0..82] OF CHAR;
47:                      | FALSE: r: RECORD
48:                                    max,akt:CHAR;
49:                                    line:String;
50:                                  END;
51:                        END
52:                  END;
53:  BEGIN
54:   ConOut(ESC); ConOut('E'); (* Schirm löschen *)
55:   ConWS('WordPlus -> TeX Converter'); ConOut(CR); ConOut(LF);
56:   ConWS(' 1988 Robert Tolksdorf'); ConOut(CR); ConOut(LF);
57:   ConOut(CR); ConOut(LF);
58:   InitStringModule;
59:   TeXStyleBegin[roman ]:='';
60:   TeXStyleBegin[bold  ]:='{\bf ';
61:   TeXStyleBegin[italic]:='{\it '; (* Für LaTeX besser : '{\em' *)
62:   TeXStyleBegin[underl]:='$\rm\underline{'; (* LaTeX : '\underline' *)
63:   TeXStyleBegin[super ]:='$^{\rm ';
64:   TeXStyleBegin[sub   ]:='$_{\rm ';
65:   TeXStyleBegin[typew ]:='{\tt ';
66:   TeXStyleEnd  [roman ]:='';
67:   TeXStyleEnd  [bold  ]:='}';
68:   TeXStyleEnd  [italic]:='\/}';
69:   TeXStyleEnd  [underl]:='}$'; (* LaTeX : '}' *)
70:   TeXStyleEnd  [super ]:='}$';
71:   TeXStyleEnd  [sub   ]:='}$';
72:   TeXStyleEnd  [typew ]:='}';
73:   Quotes[FALSE]:="``"; Quotes[TRUE]:="'''; quote:=FALSE;
74:   NewPageCommand:='\vfill\eject'; (* LaTeX : '\newpage' *)
75:   BeginCommand:=''; (*LaTeX : '\documentstyle{report} \begin{document}'*)
76:    EndCommand    :='\end';           (* LaTeX : '\end{document} *)
77:   Hyphenation:='\-'; (* Trennvorschlag *) aktTeXStyle:=roman ;
78:   CharMap[020H]:='~';  CharMap[021H]:='!'; CharMap[022H]:=Quotes[quote];
79:   CharMap[023H]:='\#'; CharMap[024H]:='\$'; CharMap[025H]:='\%';
80:    CharMap[026H]:='\&';
81:   FOR c:=027H TO 05AH DO CharMap[c,0]:=CHAR(c); CharMap[c,1]:=0C; END;
82:   CharMap[03CH]:='$<$'; CharMap[03EH]:='$>$';
83:  CharMap[05BH]:='$[$'; CharMap[05CH]:='$\backslash$';
84:   CharMap[05DH]:='$]$'; CharMap[05EH]:='\^'; CharMap[05FH]:='\_';
85:   FOR c:=060H TO 07AH DO CharMap[c,0]:=CHAR(c); CharMap[c,1]:=0C; END;
```

Listing geht weiter...

freuen, wenn Tricks zur besseren Zeichengestaltung vielleicht in einem Leserbrief veröffentlicht würden.

Die Konstanten BeginCommand und EndCommand werden am Anfang und Ende der Ausgabe erzeugt. Besonders bei der Verwendung von LaTeX wird dadurch die Konvertierung erleichtert. Die Änderungen für LaTeX sind an mehreren Stellen in den Kommentaren beschrieben. Bei weichen WordPlus-Trennungen wird ein "\-" Trennvorschlag ausgegeben. Da WordPlus teilweise recht unsinnige Stilanweisungen abspeichert, können auch Ausgaben wie "{\bf }", die keinen TeXt erzeugen, entstehen.

Dirty Tricks ....

Die zentrale Routine für das Einlesen des WordPlus-Textes ist NextWord, das ein "Stück" aus der Eingabe liest. Dabei werden Dinge wie Absatzumbruch, Stiländerung oder ein normales Wort als Ergebnis geliefert. Die Prozedur ließe sich übrigens auch für andere Arbeiten mit dem WordPlus-Format verwenden.

Das Überlesen bestimmter WordPlus-Steuersequenzen und der mehrfachen Leerzeichen geschieht durch rekursiven Aufruf und unter Weiterreichung des Parameters. Dadurch wird die Prozedur kurz und elegant.

Formatzeilen von WordPlus werden überlesen. Die Verarbeitung verschiedener Schriftgrößen wäre für TeX und LaTeX zu unterschiedlich, als daß sie hier Platz fände. Bilder werden ebenfalls ignoriert, auch wenn eine Konvertierung prinzipiell möglich wäre (s.u.).

Wem schließlich noch die Deklaration von toconvert suspekt erscheint: GEMDOS erwartet einen Record, in dem außer dem Puffer für die Eingabe mit ConRS auch noch maximale und tatsächliche Eingabelänge vermerkt sind. Die TDI-Bibliotheken erwarten als Parameter ein CHAR-Feld. Durch einen varianten Record wird der Compiler überlistet und das Programm arbeitet korrekt, auch wenn es umständlich zu notieren ist.

Das Modul DiskOps (Listing 2 und Listing 3) übernimmt die gesamte Massenspeicherein- und ausgabe. Einerseits ist eine solche Aufsplittung in Modula-2 sinnvoll, andererseits wird damit das etwas langsame Filesystem von TDI umgangen.

Reinhard Schuster Computer

OBERE MÜNSTERSTR. 33–35 · TEL. (0 23 05) 37 70 · BTX 023053770 · 4620 CASTROP-RAUXEL



Schneider COMPUTER DIVISION Vertragshandler · ATARI System-Fachhandler · star der ComputerDrucker Vertragshandler · AMSTRAD Vertragshandler · Commodore Vertrags Werkstatt

ATARI-ST-SOFTWARE

SOFTWARE FÜR ST

Titel	Preis
10Th Frame	76.60
221B Baker Street	47.90
3D Galaxy	54.10
500 CC Grand Prix	58.80
AB ZOO	47.90
Addicta Ball	47,90
Advanced Art Studio	76.60
Airball	76.60
Airball Construction Set	47.90
Altair	54.10
Amazon	54.10
American Pool	29.30
Annals of Rome	76.60
Arcade Force Four	69.90
Arena	88.30
Arkanoid	38.60
Asphai Trilogy	76.60
Asterix im Morgenland	59.90
Autoduel	58.80
Backlash	49.90
Bad Cat	55.70
Balance of Power	85.20
Bard's Tale 1	79.90
Baseball Gamestar	76.60
Battleships	47.90
Bermuda Project	83.40
Beyond the Ice Palace	67.00
Beyond Zark	74.30
Black Cauldron	76.60
Black Lamp	58.80
Blue War	54.10
Blueberry und das Gespenst	59.90
BMX Simulator	46.40
Bob Moran in Middle Age	59.90
Bob Morane Sience Fiction	58.80
Bob Winner	67,00
Bobo	59.90
Borrowed Time	59.90
Boulderdash Construction Set	76,60
Brian Clough Football	76.60
Bridge Player 2000	59.90
Bubble Bobble	58.80
Bubble Ghost	58.80
Buggy Boy	67.00
Bureaucrazy	105.40
Captain America	58.80
Captain Blood	69.90
Carrier Command	76.60
Casino Roulette	59.90
Chamonix Challenge	58.80
Championship Baseball	69.90
Championship Football	76.60
Championship Wrestling	76.60
Chopper X	29.30
Clever & Smart	58.80
Colonial Conquest	89.90
Computer Hits	89.90
Crafton & Xunk	59.90
Crash Garet	59.90
Crazy Cars	55.70
Creator	59.90
Crystal Castles	47.90
Dark Castle	69.90
Deathstrike	43.30
Deep Space	105.40
Defender of the Crown	77.40
Deflector	59.90
Deja Vu	66.60
Diablo	59.90
Dizzy Wizzard	55.70
Dungeon Master	74.30
Eco	58.80
Eden Blues	69.90
Electronic Pool	57.20
Enduro Racer	58.80
Epyx (The Worlds Greatest)	77.40
Euro Soccer 88	59.90
Extensor	33.60
Extravaganza	35.50
Eye	43.30
Fahrenheit 451	54.10
Fire Blaster	29.90
Flight Sim. Scenery Disk 11	57.20
Flight Sim. Scenery Disk 7	57.20
Flight Simulator 2	119.70
Western European Scenery	67.00
Flintstones	59.90
Football Fortunes	69.90
Football Manager 2	59.90
Formula 1 Grand Prix Sim.	55.70
Foundations Waste	69.90
Fred Feuerstein	58.80
Fright Night	55.70
Gato	85.20
Gauntlet 1	66.60
Gauntlet 2	67.00
Get Dexter 2	59.90
Giana Sisters	55.70
Gold Runner 2	58.80
Gnome Ranger	43.30
Gold Runner	69.90
Golden Path	57.20
Gold Runner 2 Scenery Disc 1	18.50
Scenery Disc 2	18.50
Guild of Thieves	69.90
Gunship	74.30
Hacker	59.90
Hacker 2	76.60
Hardball	66.60
Harrier Strike Mission	88.30
Hellowoon	69.70
Hollywood Poker	36.40
Hot Ball	69.90
I Ball	33.60
Ikari Warrior	47.90
Impact	43.30
Impossible Mission 2	55.70
Indiana Jones	49.90
Intelligent Checkers	54.10
Into the Eagles Nest	58.80
Invasion	47.90
Isnogud	58.80
Jagd auf Roter Oktober	74.30
Jewels of Darkness	59.90
Jinxter	74.30
Jump Jet	43.30
Jupiter Probe	43.30
Kaiser	123.60
Karate Kid 2	66.60
Karate Master	39.90
Karting Grand Prix	29.30
Kings Quest 3er Pack	74.30
Knightmare	59.90
Knight Orc	58.80
L'Affaire	76.60
Las Vegas	33.60
Leaderboard Golf	69.90
Leaderbord Tournament	33.60
Leatherneck	55.70
Legends of the Sword	69.90
Leisure Suit Larry	59.90
Leviathan	43.30
Liberator	39.90
Little Computer People	105.40
Livingstone	47.90
Luky Luke	
Nitroglyzerin	57.20
MacAdam Bumper	69.90
Mach 3	58.80
Marble Madness	79.90
Masters of the Universe	59.90
Mean 18 Golf	89.90
Mercenary	66.60
Metrocross	66.60
Metropolis	35.50
Mewilo	66.60
MGT	69.90
Microleague Wrestling	58.80
Mindfighter	83.40
Mindshadow	58.80
Missing One Droid	33.60
Mission Genocide	33.60
Mission Elevator	37.10
Moebius	74.30
Mortville Manor	76.60
Mouse Trap	47.90
Music Studio	105.40
Nine Princess in Amber	54.10
Ninja Mission	29.90
Nord & Bert	76.60
North Star	59.90
Not a Penny More	59.90
Obliterator	76.60
Ogre	74.30
Oids	58.80
Outcast	29.90
Out Run	55.70
Pacland	58.80
Pandora	59.90
Passengers on the Wind 2	59.90
Pengy	47.90
Perfect Match	33.60
Perry Mason	54.10
Phantasie 2	76.60
Phantasie 3	66.60
Phoenix	58.80
Pinball Factory	66.60
Pink Panther	58.80
Pirates of the Barbery Coast	39.90
Planetfall	89.90
Plundered Hearts	76.60
Plutos	43.30
Police Quest	58.80
Pool	29.30
Pool/Shuffle Board	47.90
Powerplay	59.90
Power Struggle	43.30
Protector	29.90
Psion Chess	76.60
Q Ball	57.20
Quadralian	58.80
Quantum Paint Box	59.90
Rampage	47.90
Rana Rama	58.80
Return to Genesis	58.80
Revenge 2	59.90
Rings of Zilfin	66.60
Roadrunner	66.60
Roadwar 2000	76.60
Roadwar Europe	66.60
Roadwars	59.90
Rockford	58.80
Rolling Thunder	55.70
Sapiens	58.80
Screaming Wings	47.90
Scruples	59.90
Seconds Out	55.70
Sentinel	58.80
Shackled	67.00
Shadowgate	74.30
Shanghai	69.90
Shuffleboard	29.30
Shuttle 2	66.60
Sidewinder	36.40
Silent Service	69.90
Silicon Dreams	59.90
Sky Blaster	58.80
Sky Fighter	43.30
Skyrider	58.80
Slapfight	59.90
Slaygon Adventure	55.70
Soccer Sopremo	44.80
Solomons Key	54.10
Space Ace	58.80
Space Baller	29.30
Space Port	58.80
Space Quest 1	76.60
Space Quest 2	58.80
Space Station	46.40
Spiderman	59.90
Spidertronic	59.90
Spitfire 40	69.90
Spy Versus Spy	66.60
ST Classics	59.90
ST-Soccer	55.70
ST-Wars	69.90
Stac	129.30
Staff	59.90
Star Raiders	43.30
Starglider	69.90
Startrek	58.80
Starwars	58.80
Stock Market	59.90
Stone Breaker	47.90
Strike Force Harrier	69.90
Strip Poker	59.90
Strip Poker 2	43.30
Sub Battle Simulator	66.60
Super Huey	59.90
Super Sprint	43.30
Superstar Icehockey	69.90
T.N.T.	49.90
Taipan	49.90
Tanglewood	54.10
Tau Ceti	59.90
Tee up Golf	43.30
Terramax	58.80
Terrorpods	66.60
Tetris	54.10
Thai Boxing	43.30
The Pawn	69.90
Thundercats	59.90
Time & Magic	59.90
Thrust	29.90
Time Blast	33.60
Tolteka	59.90
Tomic Tile	59.90
Tournament of Death	59.90
Tracker	74.30
Trailblazer	63.50
Trasheap	58.80
Trauma	59.90
Triviar Trove	27.80
Turbo	27.80
Turbo GT	47.90
Turbo ST	33.60
Two on Two Basketball	76.60
Ultima 3	69.90
Ultima 4	74.30
Uninvited	83.40
Universal Military Sim.	74.30
Vampires Empire	58.80
Vegas Gambler	49.90
Vermeer	77.40
War Games Contr. Set	66.60
War Hawk	29.90
Warlock's Quest	43.30
Warzone	29.90
Waterskiing	58.80
Western Games	58.80
Winter Olympiad '88	55.70
Wintergames	69.90
Wizball	58.80
Wizzards Crown	76.60
Xenon	58.80
Xevious	66.60

Telefonische Bestellung:

(02305) 3770

(Tag und Nacht)

HARDWARE

Unser Superknüller Akustikkoppler Hitrans 300 C

300 Baud, vollduplex, RS 232 C-Schnittstelle, flexibles Mittelteil, Stromversorgung über Batterie, Akku oder mitgeliefertem Netzteil möglich. Incl. FTZ-Nr. (Postzulassung), Handbuch und Netzteil.

198,–

Kunstlederhauben

260 /520 ST	17.90
520/1040 STF	21.90
Mega ST Keyboard	21.90
Mega ST Keyboard/SM 124	46.90
Mega Keyboard/SM 125	48.90
Floppy 314/354	14.90
Monitor SM 124	27.90
Monitor SM 125	29.80
Monitor SC 1224	32.90

Zweitlaufwerk 3½", 720 KB mit Netzteil 348,–

Markendisketten:

3½" 1D	24.90
3½"-2 D	29.80
Mouse-Pad	19,80
Floppy-Stecker, 14-pol.	7,90
Floppy-Kupplung 14-pol.	7.90
Monitor-Stecker 13-pol.	7,90
Monitor-Kuppl. 13-pol.	7,90

Diskettenbox 3 + 3½" für 80 3½ Disketten, abschließbar

17,90

Ladengeschäftszeiten:
Montag–Freitag 9.00 – 13.00 Uhr
15.00 – 18.30 Uhr
Samstag 9.00 – 14.00 Uhr
Langer Samstag 9.00 – 18.00 Uhr

Versand per Nachnahme zuzügl. Versandkosten. Oder Vorkasse auf Psch.-Kto. Nr. 69422-460 PschA Dortmund zuzügl. 5,– DM Versandkosten.
Ausland nur per Vorkasse auf Psch.-Kto. zuzügl. 10,– DM Versandkosten. **Bitte bei allen Bestellungen Computertyp angeben!**

Besuchen Sie unser Ladengeschäft und lassen Sie sich durch unser Fachpersonal beraten. Wir haben laufend günstige Angebote und stark reduzierte Vorführgeräte.

○ Senden Sie mir bitte Ihren Katalog (2,– DM in Briefmarken liegen bei)
○ Hiermit bestelle ich per Nachnahme:
○ Incl. kostenlosem Katalog

Vorname, Name

Straße, Hausnummer

PLZ, Ort

Datum, Unterschrift

BESTELLSCHEIN

Anz.	Artikel	Preis

Bilder konvertieren oder nicht?

Wie schon gesagt, werden Bilder ignoriert, obwohl TeX und LaTeX natürlich auch pixelorientierte Grafik ausgeben können. Ein Converter für Doodle-Bilder nach LaTeX war schon in Arbeit, allerdings stellte sich heraus, daß TeX extreme Speicherprobleme bei komplexen Bildern bekommt.

Bei der Umsetzung eines Bildes brach TeX schon in der dritten Pixel-Reihe mit einem "TeX capacy exeeded"-Fehler ab. Falls sich das Problem lösen läßt, werden Sie ein entsprechendes Programm in der ST-Computer finden.

Bequemlichkeiten

TeX als ein System, das vom Großrechner bis zum Mikro gleich implementiert ist, läßt sich mit einer grafischen Umgebung wie GEM schlecht benutzen. Die Eingabe von Parametern für ein .TTP-Programm verlangsamt den Turn-Around erheblich, und sehr lange Parametereingaben sind nicht möglich. Was TeX braucht, ist eine zeichenorientierte Umgebung, und dazu sollen Sie noch einige Tips bekommen.

Die leistungsfähigste Shell, die als Public-Domain verfügbar ist, ist der Guläm. Mit entsprechenden Batchdateien wird die Arbeit mit TeX erheblich vereinfacht.

Guläm führt die Batch-Datei "GULAM.G" beim Programmstart automatisch aus. Damit können einige TeX-Arbeitsabläufe mit dem Alias-Befehl abgekürzt werden, wie Sie in Bild 3 sehen. Nach einer Begrüßungsmeldung und der Umdefinition der Delete-Taste für den eigebauten EMACS-Editor werden einige Kommandos definiert. "dvi" ruft das DVI.PRG als GEM-Programm auf. "tex" und "lat" rufen TeX mit den entsprechenden Makropaketen auf. Hinter den Kommandos brauchen Sie nur noch den Dateinamen anzugeben.

"ed" ruft den Public-Domain-Editor PROED im Ordner "INPUTS" auf. Diese Zeile müssen Sie an den von Ihnen bevorzugten Editor anpassen. Abschließend werden noch Informationen über die neuen Kommandos angezeigt.

Zum Ausdrucken wird das Batch-File "dv" benutzt, das Sie in Bild 4 finden. "dv" erlaubt die direkte Ausgabe einer .DVI-Datei ohne den Umweg der GEM-Umgebung. Es ist auf den DVI-Treiber von TooLs-Tex abgestimmt und erhält einen Parameter, der die Druckqualität

```
 86:  CharMap[07BH]:='$\{$'; CharMap[07CH]:='$|$'; CharMap[07DH]:='$\}$';
 87:  CharMap[07EH]:='$\sim$'; CharMap[07FH]:='$\Delta$';
 88:  CharMap[080H]:='\c C';   CharMap[081H]:='ü';    CharMap[082H]:="\'e";
 89:  CharMap[083H]:='\^a';    CharMap[084H]:='ä';    CharMap[085H]:='\`a';
 90:  CharMap[086H]:='\aa';    CharMap[087H]:='\c c'; CharMap[088H]:='\^e';
 91:  CharMap[089H]:='\"e';    CharMap[08AH]:='\`e';  CharMap[08BH]:='\"\i';
 92:  CharMap[08CH]:='\^\i';   CharMap[08DH]:='\`\i'; CharMap[08EH]:='Ä';
 93:  CharMap[08FH]:='\AA';    CharMap[090H]:="\'E";  CharMap[091H]:='\ae';
 94:  CharMap[092H]:='\AE';    CharMap[093H]:='\^o';  CharMap[094H]:='ö';
 95:  CharMap[095H]:='\`o';    CharMap[096H]:='\^u';  CharMap[097H]:='\`u';
 96:  CharMap[098H]:='\"y';    CharMap[099H]:='Ö';    CharMap[09AH]:='Ü';
 97:  CharMap[09BH]:='\rlap/c'; CharMap[09CH]:='{\it \$}'; (*LaTeX: \pounds*)
 98:  CharMap[09DH]:='\rlap{Y}='; CharMap[09EH]:='ß'; CharMap[09FH]:='$f$';
 99:  CharMap[0A0H]:="\'a";    CharMap[0A1H]:="\'\i"; CharMap[0A2H]:="\'o";
100:  CharMap[0A3H]:="\'u";    CharMap[0A4H]:='\~n';  CharMap[0A5H]:='\~N';
101:  CharMap[0A6H]:='\b a';   CharMap[0A7H]:='\b o'; CharMap[0A8H]:='?`';
102:  CharMap[0A9H]:='$\lceil$';  CharMap[0AAH]:='$\rceil$';
103:  CharMap[0ABH]:='$1\over 2$'; CharMap[0ACH]:='$1\over 4$';
104:  CharMap[0ADH]:='!`';     CharMap[0AEH]:='$\ll$'; CharMap[0AFH]:='$\gg$';
105:  CharMap[0B0H]:='\~a';    CharMap[0B1H]:='\~o';  CharMap[0B2H]:='\O';
106:  CharMap[0B3H]:='\o';     CharMap[0B4H]:='\oe';  CharMap[0B5H]:='\OE';
107:  CharMap[0B6H]:='\`A';    CharMap[0B7H]:='\~A';  CharMap[0B8H]:='\~O';
108:  CharMap[0B9H]:='\"\ ';   CharMap[0BAH]:="\'\ "; CharMap[0BBH]:='\dag ';
109:  CharMap[0BCH]:='\P ';    CharMap[0BDH]:='\copyright ';
110:  CharMap[0BEH]:='\registered ';    CharMap[0BFH]:='$^{\rm TM}$';
111:  CharMap[0C0H]:='ij';  CharMap[0C1H]:='IJ';  CharMap[0C2H]:='$\aleph$';
112:   FOR c:=0C3H TO 0DCH DO CharMap[c]:=NA; END;
113:   CharMap[0DDH]:='\S ';
114:  CharMap[0DEH]:='$\wedge$'; CharMap[0DFH]:='$\infty$';
115:  CharMap[0E0H]:='$\alpha$'; CharMap[0E1H]:='$\beta$';
116:  CharMap[0E2H]:='$\Gamma$'; CharMap[0E3H]:='$\pi$';
117:  CharMap[0E4H]:='$\Sigma$'; CharMap[0E5H]:='$\sigma$';
118:  CharMap[0E6H]:='$\mu$';   CharMap[0E7H]:='$\tau$';
119:  CharMap[0E8H]:='$\Phi$';  CharMap[0E9H]:='$\Theta$';
120:  CharMap[0EAH]:='$\Omega$'; CharMap[0EBH]:='$\delta$'; CharMap[0ECH]:=NA;
121:  CharMap[0EDH]:='$\phi$';  CharMap[0EEH]:='$\in$';
122:  CharMap[0EFH]:='$\cap$';  CharMap[0F0H]:='$\equiv$';
123:  CharMap[0F1H]:='$\pm$';   CharMap[0F2H]:='$\geq$';
124:  CharMap[0F3H]:='$\leq$';  CharMap[0F4H]:=NA; CharMap[0F5H]:=NA;
125:  CharMap[0F6H]:='$\div$';  CharMap[0F7H]:='$\approx$';
126:  CharMap[0F8H]:='$\circ$'; CharMap[0F9H]:='$\bullet$';
127:  CharMap[0FAH]:='.';       CharMap[0FBH]:='$\sqrt{\ }$';
128:  CharMap[0FCH]:='$^{\rm n}$'; CharMap[0FDH]:='$^{2}$';
129:  CharMap[0FEH]:='$^{3}$'; CharMap[0FFH]:='\=\ ';
130:   SoftBlank:=' '; TeXLine:='';
131:  lastwasblank:=FALSE; lastwasdash:=FALSE;
132:   ConWS('WordPlus-Filename : ');
133:  toconvert.r.max:=CHAR(80); ConRS(toconvert.a); ConOut(CR); ConOut(LF);
134:  ConOut(CR); ConOut(LF); ConWS(toconvert.r.line); ConWS(' -> ');
135:   IF OpenFiles(toconvert.r.line) THEN
136:    ConWS(toconvert.r.line); ConOut(CR); ConOut(LF);
137:     RegisteredMacro := (* Macro für das Registered-Zeichen *)
138:  '\def\registered{{\ooalign{\hfil\hbox{\tt R}\hfil\crcr\mathhexbox20D}}}';
139:     Append(RegisteredMacro);
140:      Eject;
141:     Get(aktchar); (* erstes Zeichen einlesen *)
142:      RETURN TRUE
143:    ELSE
144:     ConWS('Fehler beim Öffnen'); ConOut(CR); ConOut(LF);
145:      RETURN FALSE
146:    END;
147: END Init;
148:
149: (* Holt nächstes Wort aus dem Wordplus-Text Ergebnis ist ein varianter
150:    Record, mit dem alle Text-Bestandteile abgedeckt werden      *)
151: PROCEDURE NextWord(VAR AWord:AWordType);
152: VAR nohyphen : BOOLEAN;
153:
154: BEGIN
155:   CASE aktchar OF
156:     ESC : AWord.Type:=StyleChange; (* Stiländerung *)
157:           Get(aktchar);
158:           IF    fett   IN BITSET(aktchar) THEN
159:             AWord.Style:=bold;
160:           ELSIF hell   IN BITSET(aktchar) THEN
161:             AWord.Style:=typew;
162:           ELSIF kursiv IN BITSET(aktchar) THEN
163:             AWord.Style:=italic;
164:           ELSIF unters IN BITSET(aktchar) THEN
165:             AWord.Style:=underl;
166:           ELSIF hochge IN BITSET(aktchar) THEN
167:             AWord.Style:=super;
168:           ELSIF tiefge IN BITSET(aktchar) THEN
169:             AWord.Style:=sub;
170:           ELSE
171:             AWord.Style:=roman;
```

Listing geht weiter...

```
172:           END;
173:           Get(aktchar);
174:     | FF  : AWord.Type:=NewPage;  (* Fester Seitenumbruch *)
175:           Get(aktchar);
176:           lastwasblank:=FALSE;
177:     | RS,GS,
178:       FS  : Get(aktchar);
179:           IF ~lastwasblank THEN (* wenn nicht schon ein Blank gelesen *)
180:             AWord.Type:=Blank;
181:             lastwasblank:=TRUE;
182:           ELSE
183:             NextWord(AWord)  (* ansonsten weiter *)
184:           END;
185:     | US  : (* Formatzeile  -> ignorieren *)
186:           WHILE (aktchar#CR) AND (~EOst()) DO  Get(aktchar);  END;
187:           Get(aktchar); (* LF überlesen *)
188:           Get(aktchar); (* nächstes Zeichen *)
189:           NextWord(AWord);
190:     | CR  : Get(aktchar); (* CR/LF *)
191:           Get(aktchar);
192:           IF ~lastwasblank THEN (* falls hartes Return -> neuer Absatz *)
193:             AWord.Type:=NewPar
194:           ELSE
195:             NextWord(AWord); (* ansonsten weiter *)
196:           END;
197:     | VT  : Get(aktchar);    (* vertical tab ignorieren *)
198:           Get(aktchar);
199:           NextWord(AWord);
200:     | CAN : REPEAT
201:             Get(aktchar);
202:           UNTIL (aktchar=CAN) OR (EOst());
203:           Get(aktchar);
204:           NextWord(AWord);
205:     ELSE  (* normales Wort *)
206:       WITH AWord DO
207:         Type:=Normal;
208:         TheWord:='';
209:         nohyphen:=FALSE;
210:         lastwasdash:=FALSE;
211:         (* auf spezielle TeX-Zeichen überprüfen und eventuell ersetzen *)
212:        WHILE (aktchar IN SetOfChar{' '..CHAR(0FFH)}) AND (~EOst()) AND
213:              (Length(TheWord)<MaxLineLength) DO
214:         Concat(TheWord,CharMap[INTEGER(aktchar)],TheWord);
215:           (* An- und Abführungszeichen schalten *)
216:           IF aktchar='"' THEN
217:             quote:=NOT quote;
218:            CharMap[INTEGER(aktchar)]:=Quotes[quote];
219:           END;
220:          lastwasdash:=(aktchar='-');
221:          Get(aktchar); (* nächstes Zeichen *)
222:           (* Trennzeichen, also noch gleiches Wort *)
223:          IF (aktchar=EM) THEN (* weiches Trennzeichen *)
224:            Concat(TheWord,Hyphenation,TheWord);
225:             Get(aktchar); (* CR überlesen *)
226:             Get(aktchar); (* LF überlesen *)
227:             Get(aktchar); (* nächstes Zeichen *)
228:           ELSE
229:            IF ((aktchar=CR) AND lastwasdash) THEN (* hartes Trennzeichen *)
230:              Get(aktchar); (* LF überlesen *)
231:              Get(aktchar); (* nächstes Zeichen *)
232:            END;
233:           END;
234:         END;
235:         lastwasblank:=FALSE;
236:       END;
237:     END;
238: END NextWord;
239:
240: PROCEDURE Eject; (* TeX-Zeile ausgeben *)
241: BEGIN
242:   Put(TeXLine);
243:   TeXLine:='';
244: END Eject;
245:
246: (* Wort an TeX-Zeile anhängen. Eventuell umbrechen und ausgeben *)
247: PROCEDURE Append(VAR w:String);
248: BEGIN
249:   IF Length(TeXLine)+Length(w)>MaxLineLength THEN
250:     WHILE (Length(TeXLine)>0) AND (TeXLine[Length(TeXLine)-1]=' ') DO
251:       TeXLine[Length(TeXLine)-1]:=0C
252:     END;
253:     Eject;
254:     TeXLine:=w;
255:     WHILE TeXLine[0]=' ' DO
256:       Delete(TeXLine,0,1); (* keine Blanks am Anfang einer Zeile *)
```

Listing geht weiter...

bestimmt.

Vorher müssen Sie für jede Qualitätsstufe im DVI.PRG die Voreinstellungen vornehmen und abspeichern. Das erzeugte DVI.INF wird dann entsprechend umbenannt. Im konkreten Fall enthält z.B. die Datei DVIPRINT.INF die Voreinstellungen für einen Ausdruck auf einem NEC-P6 in 360*360-Auflösung.

"DV.G" benennt nun entsprechend dem Parameter von 1 bis 4 diese Dateien in DVI.INF um. Dann wird DVI.PRG im Batch-Modus aufgerufen und schließlich die .INF-Datei wieder umbenannt.

Damit wird der Durchlauf einer TeX-Eingabe erheblich vereinfacht. Für den TeXt "test" würden die Eingaben lauten "ed test", "tex test" und "dv 3 test", um einen Ausdruck in bester Qualität auf einem NEC zu erhalten. Gegenüber der GEM-Umgebung ist dieses Vorgehen eindeutig effizienter.

Robert Tolksdorf

```
DEFINITION MODULE DiskOps;

FROM Strings  IMPORT String ;

PROCEDURE OpenFiles(VAR
filename:String):BOOLEAN;

PROCEDURE CloseFiles;

PROCEDURE EOst():BOOLEAN;

PROCEDURE Get(VAR ch:CHAR);

PROCEDURE Put(VAR ln:String);

END DiskOps.
```

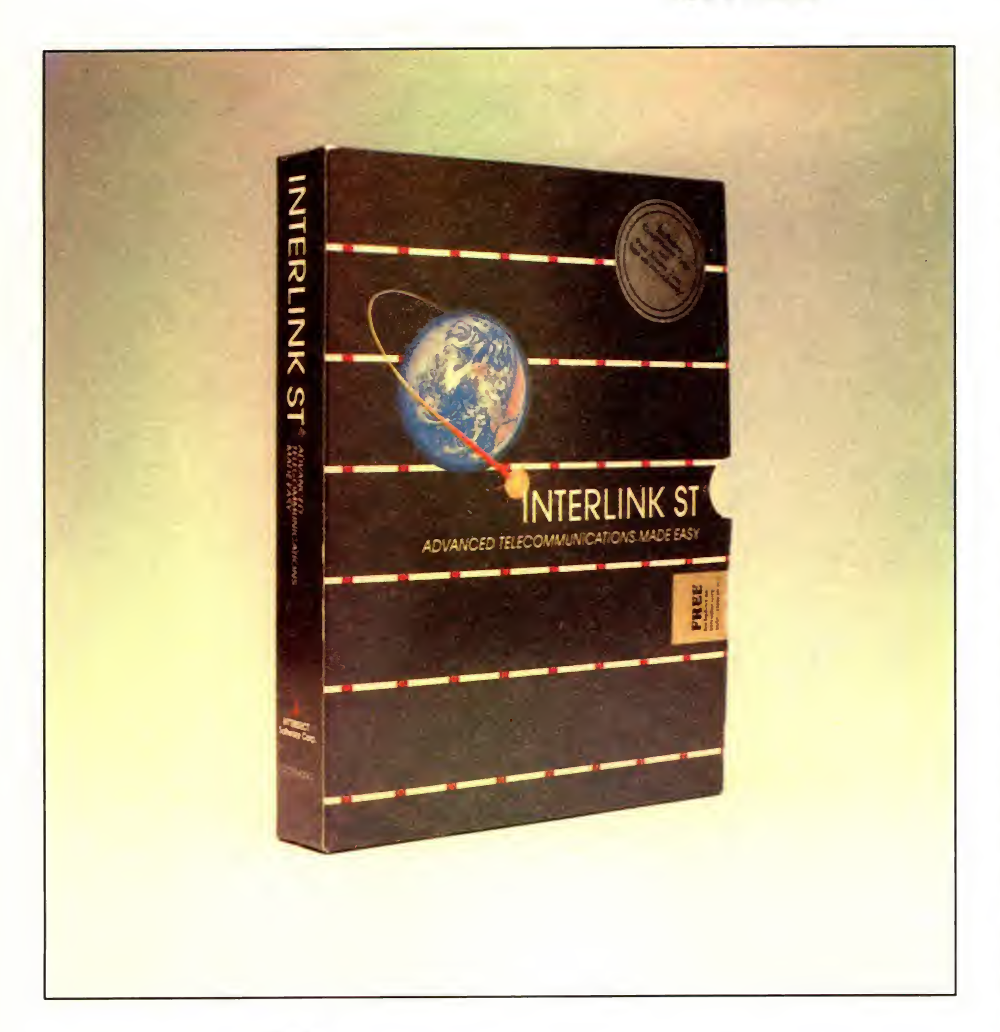
Listing 2: Das Definitionsmodul von Diskops

```
257:       END;
258:     ELSE
259:      Concat(TeXLine,w,TeXLine);
260:     END;
261:  END Append;
262:
263:  VAR myword: AWordType;
264:      lastwaspar : BOOLEAN;
265:
266:  BEGIN
267:    IF Init() THEN
268:     Append(BeginCommand);
269:      Eject;
270:     lastwaspar:=FALSE;
271:     WHILE (~EOst()) DO (* Text wortweise einlesen und Ausgabe erzeugen *)
272:       NextWord(myword);
273:       CASE myword.Type OF
274:         Normal     : Append(myword.TheWord); (* normales Wort *)
275:        | Blank      : Append(SoftBlank);          (* weiches Blank *)
276:       | StyleChange: Append(TeXStyleEnd[aktTeXStyle]); (* alt aus *)
277:                      Append(TeXStyleBegin[myword.Style]); (* neu ein *)
278:                       aktTeXStyle:=myword.Style;
279:        | NewPage    : Eject;                             (* Seitenumbruch *)
280:                       Append(NewPageCommand);
281:                        Eject;
282:        | NewPar     : IF ~lastwaspar THEN   (* immer nur eine Leerzeile *)
283:                         Eject; Eject;       (* neuer Absatz *)
284:                        END;
285:       ELSE   (* es sollte keine anderen Wort-Typen geben ... *)
286:        END;
287:      lastwaspar:=(myword.Type=NewPar);
288:     END;
289:    Append(TeXStyleEnd[aktTeXStyle]);
290:      Eject;
291:     Append(EndCommand); (* korrekt beenden *)
292:      Eject;
293:     CloseFiles; (* Files schlieβen und fertig .. *)
294:    END;
```

Listing 1: Der TeXConverter in TDI-Modula-2

```
 1:  IMPLEMENTATION MODULE DiskOps;
 2:  (*$A+*)
 3:  (* Disk-Operationen für TeXConverter
 4:     by Robert Tolksdorf *)
 5:
 6:  FROM SYSTEM    IMPORT ADR;
 7:  FROM ASCII     IMPORT CR, LF;
 8:  FROM GEMDOS    IMPORT Create, Open, Read, Write,
 9:                        Seek, SeekMode, Close, ConOut ;
10:  FROM Strings   IMPORT Length, String, Concat ;
11:
12:  CONST BlipsInterval = 20 ;
13:        BlipsMaxWidth = 40 ;
14:
15:  VAR FileLength, BytesRead,
16:      OneByte, TwoBytes                : LONGCARD;
17:      stWordHandle, TeXHandle, Blips : INTEGER;
18:       CRLF                            : String;
19:       OneBlip                         : CHAR;
20:
21:  (* Files öffnen und Länge des WordPlus-Textes
        feststellen *)
22: PROCEDURE OpenFiles(VAR filename:String):BOOLEAN;
23:  VAR dummy:LONGCARD;
24:  BEGIN
25:   Open(filename,0,stWordHandle);
26:    IF stWordHandle<0 THEN RETURN FALSE END;
27:    (* Seek an's Ende liefert Filelänge *)
28:   Seek(0,stWordHandle,end,FileLength);
29:   Seek(0,stWordHandle,beginning,dummy);
30:    WHILE filename[Length(filename)-1]#'.' DO
31:      filename[Length(filename)-1]:=0C;
32:     END;
33:   Concat(filename,'TEX',filename);
34:   Create(filename,0,TeXHandle);
35:    IF TeXHandle<0 THEN
36:     Open(filename,1,TeXHandle);
37:    END;
38:    RETURN TRUE;
39:  END OpenFiles;
40:
41:  (* Files schlieβen *)
42:  PROCEDURE CloseFiles;
43:  VAR dummy:BOOLEAN;
44:   BEGIN
45:    dummy:=Close(stWordHandle);
46:    dummy:=Close(TeXHandle);
47:   END CloseFiles;
48:
49:   (* Ende der Eingabe ? *)
50:  PROCEDURE EOst():BOOLEAN;
51:   BEGIN
52:    RETURN (BytesRead>=FileLength)
53:   END EOst;
54:
55:   (* ein Zeichen aus Eingabetext lesen *)
56:  PROCEDURE Get(VAR ch:CHAR);
57:   BEGIN
58:   Read(stWordHandle,OneByte,ADR(ch));
59:    INC(BytesRead);
60:   END Get;
61:
62:   (* eine Zeile in die Ausgabe schreiben *)
63:  PROCEDURE Put(VAR ln:String);
64:  VAR ToWrite:LONGCARD;
65:   BEGIN
66:    INC(Blips);  (* Blips ausgeben *)
67:    IF (Blips MOD BlipsInterval)=0 THEN
68:      IF (Blips MOD (BlipsInterval*BlipsMaxWidth))=0 THEN
69:        ConOut(CR);
70:        ConOut(LF);
71:      END;
72:     ConOut(OneBlip);
73:    END;
74:   ToWrite:=LONGCARD(Length(ln));
75:   Write(TeXHandle,ToWrite,ADR(ln));
76:   Write(TeXHandle,TwoBytes,ADR(CRLF));
77:   END Put;
78:
79:   BEGIN
80:    BytesRead:=0; (* einige feste Variablen wg. TDI-Lib.*)
81:    OneByte:=1;
82:    TwoBytes:=2;
83:     CRLF:='  ';
84:    CRLF[0]:=CHAR(0DH);
85:    CRLF[1]:=CHAR(0AH);
86:    Blips:=-1; (* Damit am Anfang ein Blip erscheint *)
87:    OneBlip:='*';
88:   END DiskOps.
```

Listing 3: Das Implementationsmodul von Diskops



DFÜ IM GRIFF INTERLINK ST

Interlink ST ist ein nicht mehr ganz neues, aber sehr interesssantes Telekommunikationsprogramm mit einem Leistungsumfang, der fast keinen Wunsch mehr offen läßt. Zum Test lag die Version 1.75 vor, die aber inzwischen durch Version 1.8x ersetzt wurde. Das Programm wurde von Randy Mears (Music Construction Set) geschrieben und wird in den USA von Intersect Software vertrieben. Interlink ist nicht nur ein Terminalemulator, doch dazu später mehr.

Beim Starten des Programms erscheint zunächst eine Statusseite, von der über die Menüs, Mausklicks und Befehlstasten in verschiedene Arbeitsmodi umgeschaltet werden kann. Auf der Statusseite werden außerdem die momentanen Einstellungen für die wichtigsten Parameter angezeigt. (Geschwindigkeit, Parität, Verbindungsdauer, Emulation . . .)

Mit typischen USA-Merkmalen,...

Da Interlink aus dem USA stammt, ist es natürlich (der Post sei's geklagt) Hayeskompatibel und die Modemvoreinstellungen sind dementsprechend. Falls man also nicht über ein solches Modem verfügt (darf man ja auch nicht), kann man die Modemfunktionen abstellen oder entsprechend festlegen. Sehr leistungsfähig präsentiert sich auch das Autowahl-Menü. Man kann nicht nur versuchen, ein Ziel mehrfach anzurufen, sondern auch eine Gruppe von Mailboxen auswählen, die dann wechselweise immer wieder angewählt werden, bis eine Verbindung zustandekommt. Man kann also zum Beispiel alle Kölner auf einen Teil der Wähl-Buttons legen, so man sie vorher definiert hat. Zu jedem Wähl-Button merkt sich Interlink die Nummer, ein eventuelles Präfix oder Suffix (jaja, in den USA kann man sich beim Wählen aussuchen, welcher Firma man das Gespräch bezahlen möchte) und eine automatische Einlogsequenz.

... aber auch völlig neuen Möglichkeiten,...

Die Einlogsequenz erstellt man einfach, indem man den integrierten Recorder anstellt und mitlaufen läßt, während man die Verbindung aufbaut. Später kann man diese Sequenz speichern und sie bei Bedarf entweder so schnell wie möglich oder in der Originalgeschwindigkeit abspielen. Außerdem läßt sich einstellen, wie lange die Sequenz auf eine Antwort der Gegenstelle warten soll, und wieviele der letzten Zeichen einer zu beantwortenden Frage relevant sind.

Wieder ein Problem für den geplagten Nutzer der Postdienste, diesmal DATEX-P. DATEX-P schickt beim Einloggen über Telefon immer eine 11- stellige Zahl mit, deren letzte Stellen sich unterscheiden, je nachdem welcher Kommunikationskanal gerade frei ist. Da Interlink immer die letzten Zeichen der Gegenstelle vor einer abzusendenden Eingabe prüft, gibt es hier Unstimmigkeiten, so daß diese hervorragende Möglichkeit nicht nutzbar ist.

Eine Übersetzungstabelle ermöglicht die korrekte Kommunikation mit einem deutschsprachigen Datendienst, und daß tatsächlich Umlaute auf dem Bildschirm erscheinen anstatt irgendwelcher Klammern.

Man kann natürlich auch mit einem Kollegen oder Freund eine solche Tabelle austauschen und hat so die Möglichkeit, die Kommunikation zu verschlüsseln.

Interlink verfügt, wie es bei Programmen dieser Art üblich ist, über einen edierbaren Puffer, dessen Größe sich aus bestimmten Gründen während des Betriebs nicht verändern läßt. Die Puffergröße muß voreingestellt werden, ebenso kann man einen bis zu 31K großen Schnittstellenpuffer festlegen, der für den Multitasking (!) - Betrieb benötigt wird.

geht weiter...

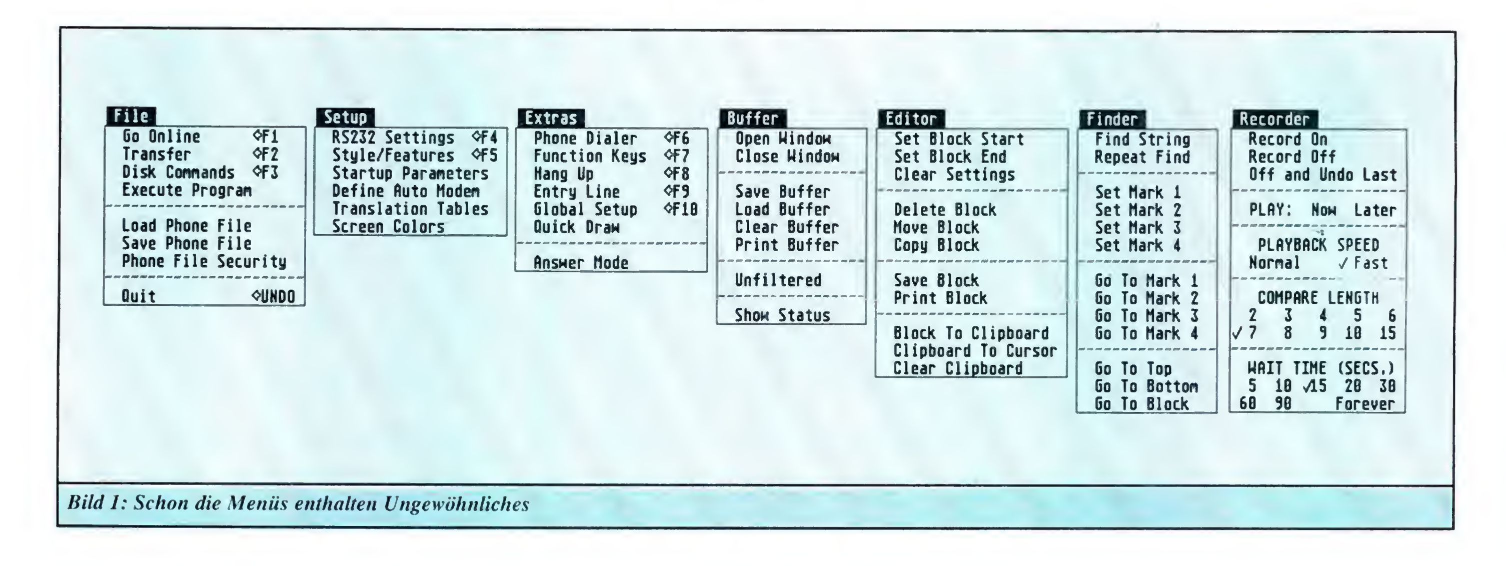


Bild 1: Schon die Menüs enthalten Ungewöhnliches

... einem kompletten Texteditor,...

Die Edierfunktionen des Puffers sind vollständig, aber zum Teil nur mit der Maus über die Menüs zu erreichen. Es lassen sich ein Block definieren, bis zu 4 Sprungmarken setzen, ein Wordwrap ist einschaltbar, und, es kommt Mac-Feeling auf, man kann einen Block in ein Clipboard schieben. Dieses Clipboard hält sich an den bisher leider noch von keinem anderen Programmhersteller verwendeten Standard, der von Digital Research für GEM vorgegeben wurde, der aber in den meisten Büchern zum ST nicht auftaucht. Das Clipboard wird in ein Systemfile abgelegt, und ist dort von einem anderen Programm, das dem Standard folgt, abrufbar. Eine sehr nützliche Eigenschaft, die hoffentlich Schule macht.

Damit das Clipboard auch genutzt werden kann, muß allerdings ein kurzes Programm namens CLIP.PRG im AUTO-Ordner installiert werden. Die Funktionstasten sind der schnellste Weg, um von einem Programmteil in den anderen zu kommen, in Zusammenhang mit der Alternate- oder der Control-Taste stehen außerdem 20 benutzerdefinierbare Tasten zur Verfügung,die entweder als Textmakros, oder zum Sequenzen aufrufen verwendbar ist.

...und Erweiterungsmöglichkeiten...

Bevor man Online geht, sollte man die Schnittstellenparameter überprüfen, wobei bis auf die Hand-shaking-Protokolle (XON/XOFF und RTS/CTS) alle Parameter auf der Menüseite angezeigt werden. Auch die Terminalemulation ist wählbar, wobei ein VT52-Emulator fest implementiert ist. Man kann auf eine Emulation verzichten, oder mit dem '?????'-Button weitere Emulationen nachladen. Zur Zeit gibt es eine VT 100/102-Emulation, eine Tektronix 4014 sowie VT 220 sind in Vorbereitung.

Zur besseren optischen Hervorhebung läßt sich die Statuszeile auf dem Online-Bildschirm auch invers darstellen und mit dem monochromen Monitor (SM124) wird auch die Nutzung des 48-Zeilen-Modus ermöglicht. Schließlich läßt sich noch ein akustisches Signal einschalten, daß den Hacker dezent darauf hinweist, daß er schon wieder eine Viertelstunde vertelefoniert hat. Bei Bedarf kann man auch für die Online Übertragung einen Wordwrap einschalten, so daß keine Zeile, die uns über Modem erreicht, länger als 80 Zeichen wird, was besonders nützlich ist, wenn die Gegenstelle z.B. 132 Zeichen pro Zeile verwendet.

Nachdem man nun alles eingestellt hat, was es einzustellen gibt, sollte man zunächst die Telefondatei sichern, da in ihr auch alle anderen Parameter mit abgelegt werden. Falls man nicht will, daß jemand anderes weiß, mit wem man kommuniziert, kann man die Telefondatei auch über ein Paßwort verschlüsseln.

Dabei wird ein Einwegalgorithmus verwendet, so daß man sich das Paßwort tunlichst sehr gut merken sollte, denn was nutzen die schönsten Daten, wenn man nicht an sie 'rankommt?

Endlich kann man mit F1 Online gehen oder eine Einlogsequenz aufrufen. Die Umschaltung zwischen den einzelnen Bildschirmen ist jederzeit möglich, wobei die einlaufenden Daten im Schnittstellenpuffer zwischengespeichert werden. Der Online-Bildschirm hat, wie bereits erwähnt, eine Statuszeile, die ganz unten auf dem Bildschirm angezeigt wird. Angezeigt wird die Uhrzeit, die Verbindungsdauer, ob die akustischen Signale aktiv sind, der Schnittstellenzustand, der Recorderstatus, die Baudrate, der Du-

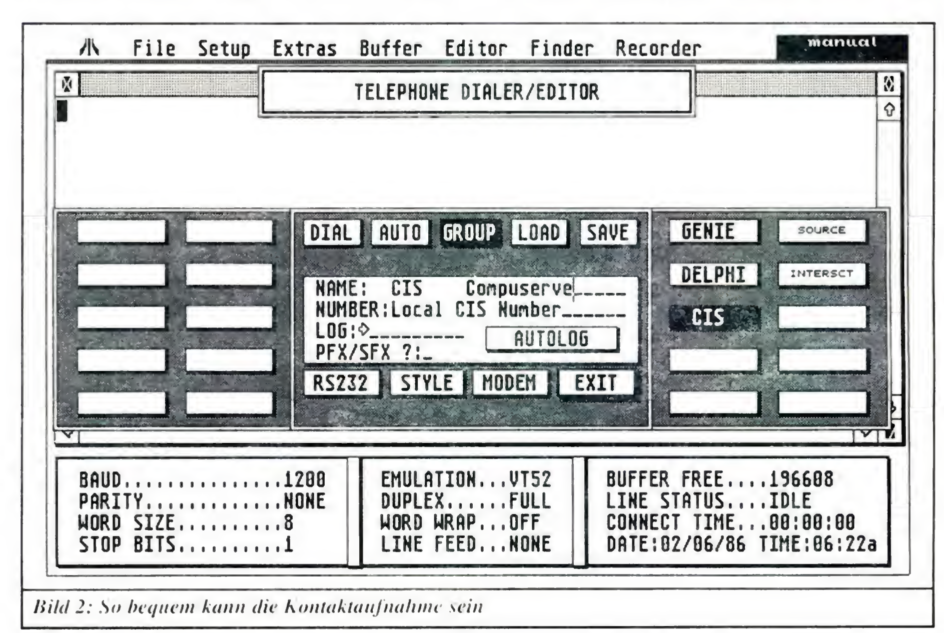


Bild 2: So bequem kann die Kontaktaufnahme sein

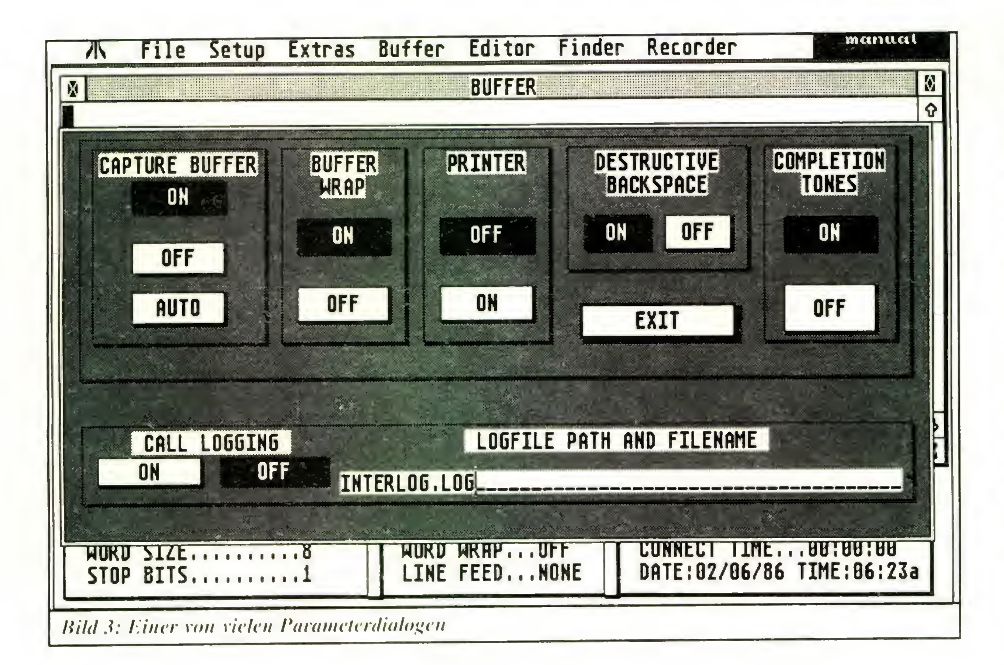


Bild 3: Einer von vielen Parameterdialogen

plexmodus und der Pufferstatus sowie der Grad seiner Füllung in Prozent. Statt der Statuszeile kann man auch einen einzeiligen Arbeitspuffer anzeigen lassen, in dem man besonders während man 'chat'tet, also sich direkt mit einem Gegenüber unterhält, schon mal die nächste Zeile zum Senden vorbereiten kann, während man oben die Antwort des Partners eintreffen sieht.

... präsentiert sich Interlink.

Interlink kennt diverse Übertragungsprotokolle für Up- und Downloads. Bereits eingebaut sind X- und Y-Modem, sowie reiner ASCII-Transfer. Auch hier sind verschiedene weitere Protokolle nachladbar. Bisher erschienen sind Compuserve Quick-B Protocol und Multitasking x(y)modem sowie Ymodem-Batch. In Vorbereitung ist neben Kermit unter anderem auch Z-Modem, das zur Zeit wohl schnellste Übertragungsprotokoll.

Bei XModem schaltet das Programm bei Bedarf automatisch zwischen den Betriebsarten "Zyklische Redundanzprüfung" (CRC) und "Prüfsumme" (CHK) um und, falls die Gegenstelle zu langsam antwortet in den slow-Modus, wo die Wartezeit zwischen den einzelnen Blöcken vergrößert wird. Bei YModem-Betrieb wird automatisch in XModem umgeschaltet, falls die Gegenstelle YModem nicht beherrscht. Der Unterschied zwischen beiden liegt nur in der Größe der einzelnen Blöcke, die übertragen werden.

Die ASCII-Übertragungen sind für jede im Wählverzeichnis enthaltene Gegenstelle getrennt einstellbar [Anforderungszeichen (sichtbar=Prompt) oder nicht, 7/8 bit, Handshake, mit oder ohne Zeilenvorschub (LF) und Wagenrücklauf (CR) sowie Verzögerungen zwischen Zeichen und Zeilen].

Nützliche Zugaben wie Multitasking...

Wie bereits mehrfach angedeutet, ist Interlink multitaskingfähig. Dazu muß neben dem MULTIXYF.TXF-Transferprotokoll ein mitgeliefertes Accessory (ILNMULTI.ACC) installiert sein. Während eines Up- oder Downloads kann man nun im Editor spielen oder ein externes GEM-Programm aufrufen (! nur <artige> Programme !). Das Accessory zeigt den Status des Up-/Downloads in seiner Menüzeile an und gibt, wenn gewünscht, akustische Signale. Up-/Downloads im Multitaskingbetrieb gehen nur vom oder auf externen Massenspeicher (Floppy/Harddisk/Ramdisk), es können weder Blöcke noch der Buffer übertragen werde. Bis 1200 baud (Höher konnte ich nicht testen.) traten auch bei längeren Übertragungen keine Probleme auf. Es ist schon ganz schön, im Vordergrund beispielsweise mit 1st Word+ zu arbeiten, während eine Übertragung läuft. Probleme gab es zum Beispiel mit Signum, da dies die Accessories ausschaltet.

...und Mini-Mailbox gehören dazu.

Als weiteres Bonbon kann Interlink auch als Mini-Mailbox genutzt werden. Es gibt einen sogenannten Answer-Modus mit 3 Zugriffsebenen, Nachrichtenbank, Up-/Download. Es gibt sogar ein Hilfsmenü. Natürlich ist Interlink kein volles Mailboxprogramm. Aber der Computer kann angerufen werden, wenn man mal für ein paar Tage weg ist, und Nachrichten entgegennehmen, die man dann selbst auch aus dem Urlaub abrufen kann. Zum Austesten dieser Funktionen gibt es auch einen Local-Button, der auf dem Bildschirm einen Anruf von außen simuliert.

Zu guter Letzt sind auch alle wichtigen

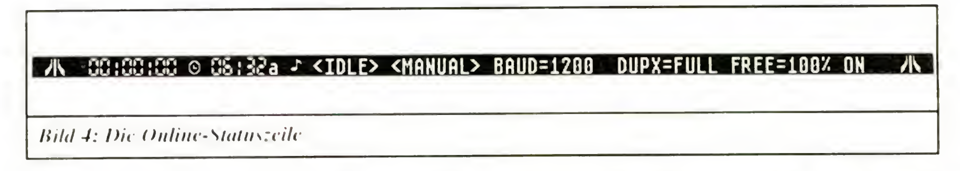


Bild 4: Die Online-Statuszeile

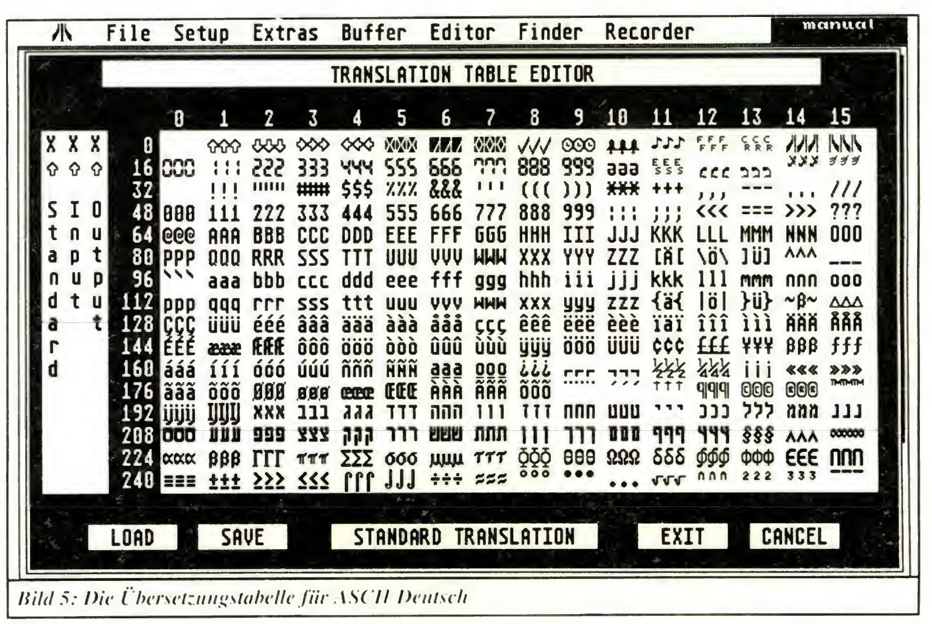


Bild 5: Die Übersetzungstabelle für ASCII Deutsch

Diskettenkommandos aus Interlink verfügbar. Disketten formatieren, Ordner anlegen, umbenennen, Dateien löschen, anzeigen und drucken sowie den Pfad einstellen sind implementiert. Im übrigen verfügt Interlink über eine eingebaute erweiterte File-Selector-Box, die die Auswahl eines anderen Laufwerks sehr vereinfacht.

Ein komplettes Paket...

Das Programm wird auf einer einseitigen Diskette geliefert und ist sowohl auf dem Monochrom- als auch auf dem Farbbildschirm lauffähig. Das Handbuch (noch (?) Englisch, 48 Seiten) ist zwar knapp aber präzise, und dank Stichwortverzeichnis und Inhaltsübersicht findet man auch schnell, was man sucht. Das Druckbild ist nicht allerbeste Qualität, aber gut lesbar. Neben den eigentlichen Funktionen des Programms wird auch noch erklärt, wie man in den USA die diversen Datendienste erreicht.

In dem Karton befinden sich auch diverse Einstiegsangebote mit erheblichen Preisnachlässen für amerikanische Datendienste wie Compuserve, The Source, Delphi, Genie und BIX. In der Summe liegen diese Nachlässe sogar über dem Verkaufspreis von Interlink. Leider haben wir hier nicht allzu viel davon, denn von Deutschland aus ist nur Compuserve direkt über einen Gateway in Frankfurt erreichbar.

Über DATEX-P erreicht man Compuserve und BIX. Genie ist zwar technisch möglich, aber General Electric, der Betreiber, hat bisher noch kein Interesse gezeigt, privaten Anwendern diesen Zugang zu öffnen.

...zuverlässig und guter Support

Im Test erwies sich Interlink als sehr zuverlässig. Bei Rechnerkopplung (Nullmodem) traten auch bei 19200 baud keine Übertragungsprobleme auf. Manchmal wirkt die VT-52-Emulation bei 300 baud etwas komisch. Das liegt aber wohl am Atari, da ähnliche Probleme auch mit anderen DfÜ-Programmen auftreten, die die interne VT-52-Emulation benutzen. Was noch fehlt, ist eine mächtige Programmiersprache, um auch ungewöhnliche Situationen definieren zu können, bei denen der Recorder nichts nützt. Sie ist für Mitte des Jahres mit Version 2.0 schon angekündigt. Was auch störte, war, daß der Puffer während des Betriebs nicht in der Größe änderbar war. Allerdings kann man mit wenigen Tastendrücken den Puffer speichern und dann leeren.

Mit Interlink erhält man ein Telekommunikationsprogramm, das nicht nur durch seine ungewöhnlichen Merkmale auffällt und besticht, sondern sich auch durch hohe Betriebssicherheit auszeichnet. Man merkt dem Programm die Reifezeit deutlich an, die ihm zuteil wurde.

CSM

Bela Computer
Unterortstr. 23-25
6236 Eschborn
Tel.: 06196/481944

ENDE

ALGORITHMEN & DATENSTRUKTUREN

Sortiermethoden B

Wie bereits in der letzten Folge angekündigt, werde ich mich heute mit den beiden Sortieralgorithmen "Heapsort" und "Quicksort" befassen. Wie sich herausstellen wird, handelt es sich dabei um Sortieralgorithmen der Ordnung O(n*log(n)) (jedenfalls im durchschnittlichen Fall). Diese Algorithmen bedeuten eine gewaltige Effizienzsteigerung gegenüber denen Algorithmen der der letzten Folge.

Heapsort

Beginnen möchte ich mit Heapsort. Sein Schöpfer, **J. Williams**, hatte dabei als grundlegenden Gedanken, daß man einen Array, entsprechende Vereinbarungen vorausgesetzt, auch als Baum interpretieren kann.

Dabei beginnt man mit dem ersten Arrayelement, das synomym mit der Wurzel verwendet wird, fährt fort mit den beiden Wurzelnachfolgern (2. und 3. Arrayelement) und sofort ...

Ergebnis ist, daß man den linken Nachfolger eines Array-, bzw. Baumelementes, durch Multiplikation des Arrayindex mit 2 erhält (2*i) und den rechten Nachfolger entsprechend durch 2*i+1.

Ein Beispiel für diesen Interpretationsvorgang finden Sie zu Beginn und Ende der Abbildung 9f.

Das Ziel ist es nun, diesem Baum eine Struktur zu geben, die Informationen über die Elementabfolge, auch über das Ausfügen eines einzelnen Elementes hinaus, bewahrt.

Dazu definierte **Williams** die Struktur des **Heaps**, genauer die des **Maximum** und des **Minimumheaps**.

Für einen Maximumheap gilt, daß jeder Vater im Baum einen größeren Schlüssel als seine (maximal) zwei Söhne besitzt. In der Arraynotation müßte also gelten:

*a[i]>=a[2*i] und a[i]>=a[2*i+1],*

soweit die a[2*i] bzw. a[2*i+1] überhaupt vorhanden sind. Entsprechend gilt für den Minimumheap:

*a[i]<=a[2*i] und a[i]<=a[2*i+1].*

Eine Eigenschaft, die sich für einen Maximumheap (Minimumheap) sofort ablesen läßt, ist die Tatsache, daß sich das größte (kleinste) Arrayelement in Position a[1] befindet. Ein Algorithmus auf der Basis der Heaps könnte also etwa folgendes Aussehen haben:

1. Überführe unstrukturierten Array in Maximumheap.
2. Solange noch Elemente im Heap vor handen.
a. Vertausche erstes und letztes Array element.
b. Füge letztes Element aus Heap aus (Dekrementiere linke Grenze).
c. Korrigiere Heapeigenschaft.

Dadurch rückt das Maximum (Minimum) an die letzte Arrayposition und ein aufsteigend (absteigend) sortiertes Array wird nach und nach von hinten aufgebaut. In der Praxis (Programmierung) sieht dies so aus, daß man eine Prozedur **korrigiere** zur Korrektur der Heapeigenschaften an einer Stelle a[i] verwendet. Dabei wird a[i] mit a[2*i] und a[2*i+1] verglichen und entsprechend den Heapanforderungen ausgetauscht.

Dies macht dann eine weitere Heapüberprüfung in dem betroffenen Teilzweig (entweder a[2*i], oder a[2*i+1]) notwendig, wofür in der Prozedur **korrigiere** (Listing 9a, Zeilen 12-41) das zentrale WHILE- Statement benutzt wird.

Wendet man nun **korrigiere** nacheinander auf die Knoten (max DIV 2) bis hin zu 1 an, erhält man aus einem unstrukturierten Array einen Heap (Heapsort, Zeilen 46-52).

Wird dann zusätzlich über alle Arrayelemente geschleift und in dieser Schleife die erste Arrayposition mit dem jeweiligen Arrayende vertauscht sowie der Heap entsprechend korrigiert (Zeilen 53-60), ist **Heapsort** fertig.

Die Wirkungsweise von **Heapsort** möchte ich Ihnen an der Abbildung 9f verdeutlichen.

Hier sehen Sie **Heapsort** bei der Arbeit. Angefangen mit einem unstrukturierten Array von sieben Elementen (1 6 2 5 9 0 7) wird in den ersten vier Schritten zunächst die Heapeigenschaft herbeigeführt.

In oben beschriebener Wechselwirkung des Ausfügens eines Elementes und der Heapkorrektur wird nun nacheinander jedes Element aus dem Heap ausgefügt, bis schließlich ein vollständig sortiertes Array erzeugt ist (0 1 2 5 6 7 9).

Ein etwas größeres Beispiel finden Sie in Abbildung 9a, hier allerdings nur mit unserer 'Schnappschußausgabe'.

Dummerweise konnte ich mich nicht motivieren, Ihnen diesen Sortiervorgang als Baumabfolge aufzuzeichnen, wofür Sie sicherlich Verständnis aufbringen werden.

geht weiter...

Quicksort (rekursiv)

Der Quicksortalgorithmus, entwickelt von Hoare, ist DER Algorithmus, der die Methode des Sortierens durch Zerlegen charakterisiert. Obwohl er sehr einfach und somit schnell zu beschreiben ist ist er in der Praxis die beste, bekannte Sortiermethode.

Erster Schritt des Quicksortalgorithmus ist die Auswahl eines beliebigen Arrayelementes, das wir als **Pivotelement** oder kurz als **pivot** bezeichnen wollen.

Alle Arrayelemente, kleiner diesem Element, werden dann auf die linke Seite des Arrays befördert, alle größeren Elemente auf die rechte Seite. Es entstehen zwei Teilarrays: a[l]..a[j] und a[i]..a[max], für die gilt:

a[x]<=a[y], für alle x aus l..j und alle y aus i..max, wobei j=i-1 ist.

Durch Anwendung von **Quicksort** auf die so entstehenden beiden Teilarrays erreicht man, indem man die Arraylänge bis auf eins herunterspielt, ein sortiertes Gesamtarray.

Realisiert finden Sie eine rekursive Variante des Quicksortalgorithmus im Listing 9b. Als **Pivotelement** wird genau das mittlere Element des Teilarrays (l..r) gewählt. Die REPEAT-Schleife realisiert die Zerlegung des Teilarrays. Zu diesem Zweck werden zwei Zeiger, i und j, von links und rechts über das Array geschoben und bei Bedarf werden zwei Elemente ausgetauscht.

Nach Abbruch dieses Vorgangs sind nur noch zwei rekursive Aufrufe, mit den neuen Teilarraygrenzen l..j und i..r zu tätigen, natürlich nur, soweit es sich dabei noch um Teilarrays handelt (l<j, bzw. i<r).

Der 'Knackpunkt' am Quicksort-Algorithmus ist die Auswahl des geigneten Pivotelementes. Hat man nämlich Pech, erwischt man ein Pivotelement, das eine sehr ungleichmäßige Aufspaltung der beiden Teilarrays zur Folge hat. Im Extremfall führt dies zu einem 1-elementigen Teilarray und einem (max-1)-elementigen Teilarray. Hält das Pech an, darf man **Quicksort** eigentlich nicht mehr als **quick** bezeichnen.

'Gut und schön, aber sehr unwahrscheinlich', werden Sie vielleicht sagen. Allerdings, ganz so unwahrscheinlich ist dieser schlechteste Fall nicht, wie das Beispiel zeigt:

```
54 22 66 89 72 58 11 49 30 96 93 91 44 82 65
54 22 66 89 72 58 82 49 30 96 93 91 44 11 65
54 22 66 89 72 91 82 49 30 96 93 58 44 11 65
54 22 66 89 96 91 82 49 30 72 93 58 44 11 65
54 22 66 89 96 91 82 49 30 72 93 58 44 11 65
54 22 91 89 96 66 82 49 30 72 93 58 44 11 65
54 96 91 89 93 66 82 49 30 72 22 58 44 11 65
96 93 91 89 72 66 82 49 30 54 22 58 44 11 65
93 89 91 65 72 66 82 49 30 54 22 58 44 11 96
91 89 82 65 72 66 11 49 30 54 22 58 44 93 96
89 72 82 65 54 66 11 49 30 44 22 58 91 93 96
82 72 66 65 54 58 11 49 30 44 22 89 91 93 96
72 65 66 49 54 58 11 22 30 44 82 89 91 93 96
66 65 58 49 54 44 11 22 30 72 82 89 91 93 96
65 54 58 49 30 44 11 22 66 72 82 89 91 93 96
58 54 44 49 30 22 11 65 66 72 82 89 91 93 96
54 49 44 11 30 22 58 65 66 72 82 89 91 93 96
49 30 44 11 22 54 58 65 66 72 82 89 91 93 96
44 30 22 11 49 54 58 65 66 72 82 89 91 93 96
30 11 22 44 49 54 58 65 66 72 82 89 91 93 96
22 11 30 44 49 54 58 65 66 72 82 89 91 93 96
11 22 30 44 49 54 58 65 66 72 82 89 91 93 96
```

Abbildung 9a: Heapsort

Als **pivot** wählen wir das erste Element des Teilarrays. Wenden wir nun den Algorithmus auf ein schon sortierte oder teilsortierte Array an, passiert folgendes:

Im Falle eines schon sortierten Arrays geschieht gerade die Katastrophe. **Quicksort** zerlegt jedesmal den Array in ein 1-elementiges Teilarray und Rest. Im Falle eines teilsortierten Arrays dürfte auch nichts besonders Anschauliches herauskommen.

Gerade diesem Mißstand beugen wir in unserer Implementierung 9b vor, indem wir immer das mittlere Element eines Arrays als Pivotelement wählen. Nichtsdestotrotz kann natürlich auch hier die Katastrophe eintreten ...

Für ganz Vorsichtige empfiehlt es sich deshalb, aus einer gewissen Anzahl von Arrayelementen das Mittlere auszuwählen. In diesem Fall dürfte Murphy eindeutig verloren haben. Man sollte hierbei allerdings darauf achten, daß die Anzahl der Auswahlelemente nicht zu groß wird, weil das wieder den Algorithmus bremst.

Nun noch ein Beispiel (Abbildung 9b). Sie sehen **Quicksort** angewendet auf unsere Standardfolge.

Das Pivotelement habe ich jeweils mit einem 'v' gekenn-zeichnet. Die Teilarrays sind durch '|' voneinander abgeschottet, und die überstrichenen Teilarrays '_' kennzeichnen den Teil der Arbeit, der bereits abgeschlossen ist.

Quicksort (iterativ)

Wegen der enormen Wichtigkeit des Quicksortalgorithmus gebe ich Ihnen noch eine iterative (nicht rekursive) Realisierung dieses Algorithmus an. Diese würde es beispielsweise auch den BASIC-Programmierern erlauben, mit den spärlichen Mitteln der Sprache ihrer Wahl **Quicksort** zu realisieren.

Im Unterschied zur rekursiven Variant haben wir nun den Stack für die Parameter (linke und rechte Arraygrenze) selbst zu verwalten.

Halt, da war doch was? Richtig, Stacks kennen wir bereits aus der ersten Folge! Was also liegt näher, als die Routinen von damals zu benutzen. Kramen Sie also am besten noch einmal Ihre ST 10/87' raus und lesen da nach ! Fertig ? Gut!

Die **stack_data** bestehen heute, entgegen den damaligen Ausführungen, nicht aus

```
                        v
54 22 66 89 72 58 11 49 30 96 93 91 44 82 65
        v
44 22 30 49 11 | 58 72 89 66 96 93 91 54 82 65
  v
11 22 30 | 49 44 | 58 72 89 66 96 93 91 54 82 65
________   v
11 22 30 | 49 44 | 58 72 89 66 96 93 91 54 82 65
________________             v
11 22 30 | 44 49 | 58 72 89 66 96 93 91 54 82 65
________________             v                ___
11 22 30 | 44 49 | 58 72 89 66 65 93 91 54 82 | 96
________________       v                      ___
11 22 30 | 44 49 | 58 54 65 | 66 89 93 91 72 82 | 96
____________________    v                     ___
11 22 30 | 44 49 | 54 | 58 65 | 66 89 93 91 72 82 | 96
______________________________      v          ___
11 22 30 | 44 49 | 54 | 58 | 65 | 66 89 93 91 72 82 | 96
______________________________      v         _______
11 22 30 | 44 49 | 54 | 58 | 65 | 66 89 82 91 72 | 93 | 96
______________________________    v           _______
11 22 30 | 44 49 | 54 | 58 | 65 | 66 72 82 | 91 89 | 93 | 96
_________________________________________ v  ______
11 22 30 | 44 49 | 54 | 58 | 65 | 66 72 | 82 | 91 89 | 93 | 96
_______________________________________________________
11 22 30 | 44 49 | 54 | 58 | 65 | 66 72 | 82 | 89 | 91 | 93 | 96
```

Abbildung 9b: Quicksort

einem Integer, sondern gleich aus einem RECORD von zwei Integern, der die jeweiligen Arraygrenzen, l und r, aufnimmt. Außerdem muß die Operation **top**, prozetural formuliert werden, weil unsere Stackdaten nicht mehr von einem einfachen Typ (sondern RECORD) sind. Die notwendigen Änderungen finden Sie anliegend.

In **Quicksort** selbst (Listing 9c) sind kaum Änderungen notwendig. Zunächst müssen zwei Variablen, x vom Typ **stack** und **el** vom Typ **stack_data**, in die Variablendeklaration aufgenommen werden. Die rekursive Subprozedur **sort** entfällt, da ihre Arbeit von der Hauptprozedur übernommen wird. Stattdessen werden, zu Beginn der Prozedur **Quicksort**, die maximalen Arraygrenzen, 1..max, auf den Stack abgelegt. Zu den Anweisungen von **Quicksort (rekursiv)**, gesellt sich nun noch eine alles umschließende REPEAT-Schleife, die solange zu durchlaufen ist, bis der **Parameterstack** leer wird. Vor Ausführung des Kernstückes (Zerlegung) von **Quicksort** sind noch die Parameter vom Stack zu holen. Nach Ausführung der Zerlegung ist statt den beiden rekursiven Aufrufen, ein Abstacken der jeweiligen Arraygrenzen notwendig.

Und das ist auch das generelle Strickmuster, mit dem rekursive Prozeduren in iterative Prozeduren & Stacks umgewandelt werden.

A: Ersetze jeden Aufruf der Funktion durch ein Abstacken der zugehörigen Parameter (push).

B: Hole, vor Ausführung der Operation die Daten vom Stack (top, pop).

C: Schließe die eigentliche Funktion durch eine REPEAT ... UNTIL is_empty(x); Schleife ein.

Als Bedingung für das Gelingen dieser Umwandlung ist allerdings gelten, daß die umzuwandelnden rekursiven Operationen mit den rekursiven Aufrufen abgeschloßen werden. Dies kann man bei den meisten, zumindest bei den einfacheren rekursiven Operationen, durch leichte Umformungen erreichen.

Noch einmal Quicksort oder Bestsort

Einer der Nachteile sämtlicher 'höherer' Sortierverfahren, zu denen auch der gerade behandelte Quicksortalgorithmus zählt, ist ihr schlechtes Verhalten bei kleinen Arrays. (Unter einem kleinen Array verstehe ich hier eines mit einer Größe weit unter 100, beispielsweise 20.) Direkt läßt sich diese Tatsache leider sehr schlecht nachprüfen, da die auftretenden zeitlichen Größen bei kleinen Arrays und dermaßen 'quicken' Sortiermethoden weit außerhalb unseres Zwei-Sekunden-Systemzeittaktes liegen.

Indirekt kann man die Zeitunterschiede allerdings gut nachprüfen, indem man einen Quicksortalgorithmus entwickelt, der bei Arrays unterhalb einer gewissen Länge auf ein 'einfaches' Sortierverfahren, beispielsweise das direkte Einfügen, umschaltet. Dieser Algorithmus sollte dann bei großen Arrays eine meßbare positive Abweichung vom normalen **Quicksort** zeigen, da eine höhere Methode 'fürs Grobe' benutzt wird und in einem Bereich, wo diese Methode nicht mehr so günstig ist, umgeschaltet wird. Ein angenehmer Nebenaspekt dieser Überlegung ist die Tatsache, daß man dadurch eine Sortiermethode erhält, die schneller als das reine Sortieren mit **Quicksort** ist, was ich durch die Verwendung der Bezeichnung **Bestsort** unterstreiche.

Will man diese Überlegungen in unsere Sortiermethode 9b einbeziehen, werden hier folgende Änderungen notwendig:

Aufbauend auf der rekursiven Prozedur **sort** ist bei den Aufrufen von **sort**, wenn tritt eine Unterschreitung eines gewissen Längen-schwellwertes (**toggle**) auftritt, in eine einfache Sortiermethode zu verzweigen. Im Listing 9d habe ich dazu **direktes_einfuegen** gewählt, was sich vom Listing 8a, der letzten Folge, lediglich durch flexible Arraygrenzen unterscheidet. Für die letzendliche Geschwindigkeit ist die Größe von **toggle** von entscheidender Bedeutung. Meine Messungen haben ergeben, daß ich mit einem Wert von 20 sehr gut liege. Das ist auch der Wert, der den Zeitmessungen zu Grunde liegt.

Um bei unserem Beispielarray diese spezielle Sortiermethode zur Geltung zu bringen, müssen wir mit **toggle** auf 5 runtergehen, da bei **toggle** = 20 und einer Arraylänge von 15 das Verfahren ein reines **direktes_einfuegen** wäre. In der Abbildung 9c sehen Sie das daraus resultierende Verhalten:

Beginnend mit einer Partitionierung des Gesamtarrays, zerfällt unser Beispiel gleich in zwei Teile, die unterschiedliches Verhalten aufweisen. Im ersten Teil wird

Mein Gott, jetzt hat er's...!

Es passiert jedem einmal: Mitten in der Arbeit am ST fehlt eine bestimmte Information. Eine fürchterliche Sucherei beginnt. Irgendwann, meist Stunden später – der befreiende Schlag an die Stirn: „Jaaa, genau." Ersparen Sie sich doch diese unnötige Aufregung:



Der DATA BECKER Führer zum ATARI ST – der ST-Anwender nennt ihn liebevoll „sein Mini-Intern". Doch das ist nicht weiter verwunderlich: In welcher Krise er sich mit seinem ST auch befinden mag, dieser DATA BECKER Führer hilft ihm weiter. Ob zur Hardware, zum Betriebssystem oder zu GEM – zuverlässig und schnell hat er hier noch jede Information gefunden. Immer auf einen Blick.
Der DATA BECKER Führer zum ATARI ST
240 Seiten, DM 29,80



Probleme mit den 1st-Programmen? Ein Blick in den DATA BECKER Führer zu 1st Word, und schon kann die Arbeit weitergehen. Ganz gleich, ob Sie mit der englischen oder der deutschen Version arbeiten, hier finden Sie alle wichtigen Informationen. 1st Word, 1st Word Plus, 1st Lektor, 1st Mailmaster oder 1st Spooler – egal, worüber Sie mehr wissen wollen, hier finden Sie die Antwort.
Der DATA BECKER Führer zu 1st Word
192 Seiten, DM 24,80



Für alle GFA-Programmierer: Der DATA BECKER Führer zu GFA-BASIC. Schnell und gezielt läßt sich mit ihm jeder Syntax-Error überprüfen. Denn hier finden Sie übersichtlich geordnet und gut strukturiert die komplette Befehlsübersicht zum GFA-BASIC. Inklusive aller nötigen Parameter und Syntax-Erläuterungen – natürlich auch zu der neuen Version 3.0!
Der DATA BECKER Führer zu GFA-BASIC
254 Seiten, DM 24,80



erscheint ca. 8/88

Der DATA BECKER Führer zu SIGNUM hilft weiter: Installation, Laden, Grafiken, Spaltensatz, Fußnoten, Tastenprogrammierung, Funktionstastenbelegung, Druckertreiber, Fonteditor – sollten Sie mal nicht weiterwissen, hier finden Sie die Lösungen. Natürlich auch mit Auflistung aller SIGNUM!-Fehlermeldungen.
Der DATA BECKER Führer zu SIGNUM!
ca. 200 S.
DM 29,80

EINSENDEN AN: DATA BECKER · MEROWINGERSTR. 30 · 4000 DÜSSELDORF 1

Hiermit bestelle ich für meinen ST

Name, Vorname

Straße

Ort

zzgl. DM 5,- Versandkosten unabhängig von der best. Stückzahl

DATA BECKER
Merowingerstr. 30 · 4000 Düsseldorf · Tel. (0211) 310010

mit **direktes_einfuegen** sortiert und im zweiten Teil mit **Quicksort** weitergearbeitet. So fortgesetzt, ergibt sich, unter Verwendung der jeweiligen Sortiermethode, ein sortiertes Gesamtarray. Die, für die Sortierung nicht wichtigen Arrayelemente, habe ich diesmal, zwecks Übersichtlichkeit, entfernt.

Testumgebung

Wie ich letztes Mal bereits erwähnte, wurde die Testumgebung Listing 8f ebenfalls hinsichtlich der heutigen Sortierverfahren entwickelt. Dummerweise ergeben sich gut meßbare Sortierzeiten bei den heutigen Prozeduren erst bei Elementanzahlen in einer Größenordnung von 10000, was im Listing 8f folgende Änderungen zur Folge hat:

Die ARRAY-Grenze für den ARRAY **a** in **field** ist auf mindestens 64000 hochzusetzen. Zusätzlich muß noch die Compileroption {$I+} vor Beginn des PROGRAM- Statements der Testumgebung gesetzt werden, womit wir unserem Pascal+-Compiler mitteilen, daß wir die Bezeichnung **Integer** synonym für **Long_Integer** verwenden möchten. Diese Änderung wird notwendig, weil die normalerweise zur Indizierung der Arrays benutzten einfachen (!) Integerwerte

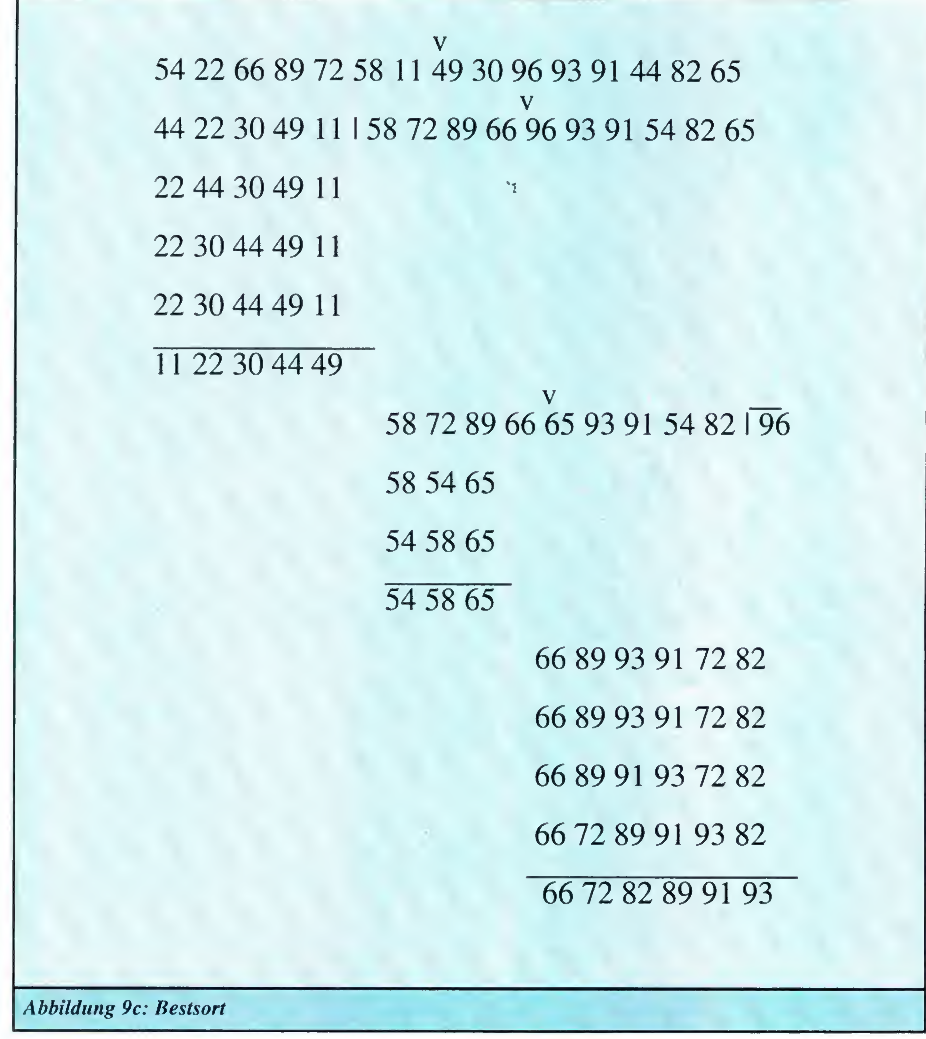


Abbildung 9c: Bestsort

Abbildung 9e: Zeitkennlinie (linear)

1. Random-Array

Elemente	Heapsort	Quicksort#1	Quicksort#2	Bestsort
8000	8	6	6	4
16000	18	12	14	10
32000	38	22	30	20
64000	78	52	68	44

2. Aufsteigend sortiertes Array

Elemente	Heapsort	Quicksort#1	Quicksort#2	Bestsort
8000	8	2	4	2
16000	18	6	8	4
32000	38	12	16	8
64000	82	28	36	20

3. Absteigend sortiertes Array

Elemente	Heapsort	Quicksort#1	Quicksort#2	Bestsort
8000	8	4	4	2
16000	16	6	8	4
32000	36	12	18	10
64000	76	28	38	22

Abbildung 9d: Sortierzeitmessungen (in Sekunden)

den Testarray nicht mehr überdecken können, was unfairerweise in einer Endlosschleife enden würde. Ästhetiker sollten auch noch die entsprechenden Hinweise auf Elementbeschränkungen im Programm ändern.

Vergleiche

Unter all diesen Änderungen ergeben sich für die vier Algorithmen der heutigen Folge die Sortierzeiten der Abbildung 9d:

Heapsort schneidet hier vergleichsweise schlecht ab, ist aber immerhin noch enorm schnell, verglichen mit den letztmaligen Prozeduren.

Quicksort (rekursiv) besitzt einen leichten Vorteil gegenüber **Quicksort (iterativ)**, was wohl durch den geringeren Aufwand für die Selbstverwaltung des Parameterstacks zu erklären ist.

Bestsort stellt alles in den Schatten und erweist sich als würdige Verbesserung von **Quicksort**.

Anzumerken ist ebenfalls noch, daß bei keinem meiner Testläufe mit Quicksort die mögliche Katastrophe, also eine sehr lange Sortierzeit, stattgefunden hat. Die graphische Auswertung der gemessenen Werte finden Sie in Abbildung 9e. Bemerkenswert ist, daß sämtliche Algorith-

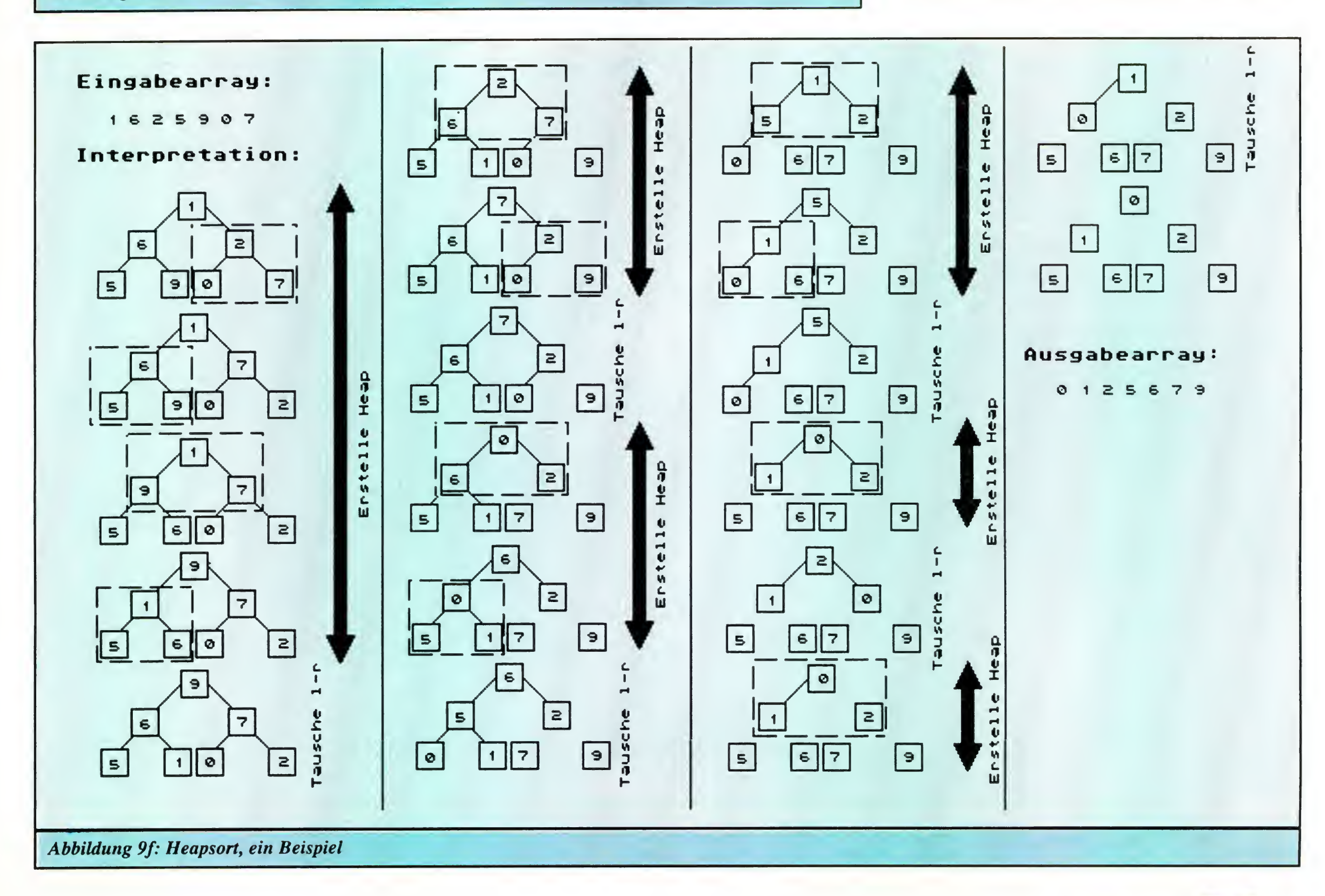


Abbildung 9f: Heapsort, ein Beispiel

men eine kaum meßbare, allerdings vorhandene, Steigung im Verhältnis Elemente/Zeit aufweisen. Eine genauere Meßmethode oder einige mathematische Überlegungen, die ich Ihnen und mir an dieser Stelle ersparen möchte, würden ergeben, daß alle vier Algorithmen eine Ordnung von O(n*log(n)) besitzen. Dabei sind die Quicksortalgorithmen noch einmal gesondert zu betrachten, da die oben erwähnte Katastrophe, also die Abspaltung in sehr kleine und sehr große Teilarrays bei der Zerlegung, die Ordnung $O(n^2)$ besitzen würde. Daher auch die Bezeichnung 'unechter' O(n*log(n)) in Abbildung 9e. Dieser schlechteste Fall kommt in der Praxis aber so gut wie nie vor, wie die untenstehende Überlegung zeigt.

Die Wahrscheinlichkeit, bei Quicksort immer das schlechteste Pivotelement auszuwählen, berechnet sich nämlich bei einem reinen Zufallsarray gerade zu 1/n!. Diejenigen unter Ihnen, die die Fakultätsfunktion kennen, wissen, daß bereits bei sehr kleinen n n! sehr groß und 1/n! darum sehr klein wird. So ergibt sich bereits bei elf Elementen eine Wahrscheinlichkeit für den schlechtesten Fall, die weit unter der von einem Lottosechser liegt (0.0000025052 %). Also auch kein besonderer Wermutstropfen im Quicksortkelch.

Vorausschau

In der letzten Folge von Algorithmen & Datenstrukturen werde ich Sortierverfahren vorstellen, die für das Sortieren auf externen Medien geeignet sind, teilweise aber auch brauchbare interne Verfahren darstellen.

Begriffe in diesem Kontext sind das 'Sortieren mit Bändern', sowie Mergesort.

Sig

```
 1: { Listing 9a   : Sortieren mit Heapsort.
 2:   programmiert : November '87
 3:   von          : Dirk Brockhaus
 4:   mit          : PASCAL+ (CCD)
 5: }
 6:
 7: PROCEDURE heapsort;
 8:
 9:    VAR l    ,
10:        r    : integer;
11:        help : key_type;
12:
13:   PROCEDURE korrigiere;
14:
15:     VAR i  ,
16:         j  : integer;
17:         ok : boolean;
18:
19:     BEGIN {korrigiere}
20:       WITH feld DO
21:         BEGIN
22:           i:=l;
23:           j:=2*i;
24:           help:=a[i];
25:           ok:=true;
26:           WHILE (j<=r) AND ok DO
27:             BEGIN
28:               IF j<r THEN
29:                 IF a[j]<a[j+1] THEN
30:                   j:=j+1;
31:               IF help>=a[j] THEN
32:                 ok:=false
33:               ELSE
34:                 BEGIN
35:                   a[i]:=a[j];
36:                   i:=j;
37:                   j:=2*i;
38:                 END;
39:             END;
40:           a[i]:=help;
41:         END;
42:     END;  {korrigiere}
43:
44:   BEGIN {heapsort}
45:     WITH feld DO
46:       BEGIN
47:         l:=(max DIV 2)+1;
48:         r:=max;
49:         WHILE l>1 DO
50:           BEGIN
51:             l:=l-1;
52:             korrigiere;
53:           END;
54:         WHILE r>1 DO
55:           BEGIN
56:             help:=a[l];
57:             a[l]:=a[r];
58:             a[r]:=help;
59:             r:=r-1;
60:             korrigiere;
61:           END;
62:       END;
63:   END;  {heapsort}
```

```
 1: { Listing 9b   : Sortieren mit Quicksort (rekursiv).
 2:   programmiert : November '87
 3:   von          : Dirk Brockhaus
 4:   mit          : PASCAL+ (CCD)
 5: }
 6:
 7: PROCEDURE quicksort_1;
 8:
 9:   PROCEDURE sort(l ,
10:                  r : integer);
11:
12:     VAR i     ,
13:         j     : integer;
14:         help  ,
15:         pivot : key_type;
16:
17:     BEGIN {sort}
18:       WITH feld DO
19:         BEGIN
20:           pivot:=a[(l+r) DIV 2];
21:           i:=l;
22:           j:=r;
23:           REPEAT
24:             WHILE a[i]<pivot DO
25:               i:=i+1;
26:             WHILE pivot<a[j] DO
27:               j:=j-1;
28:             IF i<=j THEN
29:               BEGIN
30:                 help:=a[i];
31:                 a[i]:=a[j];
32:                 a[j]:=help;
33:                 i:=i+1;
34:                 j:=j-1;
35:               END;
36:           UNTIL i>j;
37:         END;
38:       IF l<j THEN
39:         sort(l,j);
40:       IF i<r THEN
41:         sort(i,r);
42:     END;  {sort}
43:
44:   BEGIN {quicksort_1}
45:     sort(1,feld.max);
46:   END;  {quicksort_1}
```

```
 1: { Listing 9c   : Sortieren mit Quicksort (iterativ).
 2:   programmiert : November '87
 3:   von          : Dirk Brockhaus
 4:   mit          : PASCAL+ (CCD)
 5: }
 6:
 7: PROCEDURE quicksort_2;
 8:
 9:    VAR i     ,
10:        j     ,
11:        l     ,
12:        r     : integer;
13:        help  ,
```

Listing geht weiter...

3.0



Schumann
ATARI ST
GFA BASIC 3.0 FÜR EIN-STEIGER
DATA BECKER
ATARI ST
DAS GROSSE GFA BASIC BUCH
DATA BECKER
Das große GFA-Programmier-Handbuch
ATARI ST
Liesert Linden
TOOLS & ALGORITHMEN
DISKETTE IM BUCH
GFA 2.0 GFA 3.0
DATA BECKER
GFA-BASIC – ohnehin eine der leistungsstärksten BASIC-Versionen für den ST – ist jetzt in der neuen, von allen ST-Anwendern sehnlichst erwarteten Version 3.0 erhältlich. Neue Strukturanweisungen, Folding-Technik, zahlreiche Schleifenbefehle und eine höhere Geschwindigkeit ermöglichen nun eine noch effektivere Programmierung. Eine ideale Einführung in den gesamten Leistungsumfang dieses Interpreters und seiner Vorgänger findet der Anfänger in „GFA für Einsteiger". Vom einfachen Start bis hin zum ersten GEM-Projekt wird hier alles ausführlich und anhand praktischer Beispiele erläutert: Sprünge und Schleifen, Daten und Operationen, Prozeduren und Funktionen mit Parametern, Daten-Bearbeitung, Diskettenzugriff, Grafik und, und, und. Dazu die nötigen Tips für die Anpassung von GFA-2.0- auf GFA-3.0-Programme! GFA für Einsteiger hat ca. 250 Seiten und wird ab ca. 8/88 für DM 29,– erhältlich sein.
Wer noch tiefer in die professionelle Programmierung mit GFA-BASIC vorstoßen will, sollte mit dem großen GFA-BASIC-Buch arbeiten. Dieses Buch verleiht Ihnen den letzten Schliff mit all den nötigen Tips und Tricks. Anhand einer RAM-internen Dateiverwaltung lernen Sie diese mächtige Interpreter-Sprache professionell zu nutzen. And-Or-Verknüpfungen, mehrzeilige Suchvorgabe, Sortieralgorithmus, Nutzung der AES-Routinen zur Window- und Objektprogrammierung – alles, was Sie für eine professionelle Programmentwicklung unter GFA brauchen, können Sie hier praxisnah lernen und anwenden. Natürlich auch in diesem Buch eine ausführliche Beschreibung des GFA-Compilers und eine komplette Befehlsübersicht der Version 3.0! Das große GFA-BASIC-Buch – in der Neuauflage mit knapp 600 Seiten ab ca. 8/88 für DM 49,– erhältlich.
Nun könnte man meinen, wer mit dem großen GFA-BASIC-Buch gearbeitet hat, ist in Sachen GFA-BASIC absolut fit. Stimmt auch! Doch warum das Rad nochmal erfinden? Das große GFA-Programmierhandbuch bietet Ihnen für die eigentliche Programmentwicklung Lösungen, die Sie für Ihr Programm sofort verwenden können. So finden Sie hier die wichtigsten Standardalgorithmen aus nahezu allen Bereichen. Ob Sortier-Routinen oder Verketten von Feldern, ob bestimmte Algorithmen zur Textverarbeitung oder Dateiverwaltung, zu Grafikprogrammen oder zu mathematischen Berechnungen, auf über 400 Seiten zeigt Ihnen dieses Buch, wie Sie solche Probleme richtig angehen. Auf einer mitgelieferten Diskette finden Sie zusätzlich noch eine Reihe nützlicher Tools wie einen REM-Killer für GFA-BASIC oder ein Tool für ein automatisches Backup-System. Für DM 59,– können Sie sich somit jede Menge Ärger beim Programmieren sparen.
DATA BECKER
Merowingerstr. 30 · 4000 Düsseldorf · Tel. (0211) 310010
EINSENDEN AN: DATA BECKER · MEROWINGERSTR. 30 · 4000 DÜSSELDORF 1
Hiermit bestelle ich für meinen ATARI ST
Name, Vorname
Straße
Ort
zzgl. DM 5,– Versandkosten unabhängig von der best. Stückzahl

Fortsetzung Listing 9c

```
14:        pivot : key_type;
15:
16:         x     : stack;
17:        el    : stack_data;
18:
19:   BEGIN {quicksort_2}
20:     create(x);
21:     el.l:=1;
22:     el.r:=feld.max;
23:     push(x,el);
24:     WITH feld DO
25:       REPEAT
26:         top(x,el);
27:         l:=el.l;
28:         r:=el.r;
29:         pop(x);
30:         i:=l;
31:         j:=r;
32:        pivot:=a[(l+r) DIV 2];
33:         REPEAT
34:           WHILE a[i]<pivot DO
35:             i:=i+1;
36:           WHILE pivot<a[j] DO
37:             j:=j-1;
38:           IF i<=j THEN
39:             BEGIN
40:               help:=a[i];
41:               a[i]:=a[j];
42:               a[j]:=help;
43:               i:=i+1;
44:               j:=j-1;
45:             END;
46:         UNTIL i>j;
47:         IF l<j THEN
48:           BEGIN
49:             el.l:=l;
50:             el.r:=j;
51:             push(x,el);
52:           END;
53:         IF i<r THEN
54:           BEGIN
55:             el.l:=i;
56:             el.r:=r;
57:             push(x,el);
58:           END;
59:       UNTIL is_empty(x);
60:   END;  {quicksort_2}
```

```
34:
35:   PROCEDURE sort(l ,
36:                  r : integer);
37:
38:     VAR i     ,
39:         j     : integer;
40:         help  ,
41:         pivot : key_type;
42:
43:     BEGIN {sort}
44:       WITH feld DO
45:         BEGIN
46:           pivot:=a[(l+r) DIV 2];
47:           i:=l;
48:           j:=r;
49:           REPEAT
50:             WHILE a[i]<pivot DO
51:               i:=i+1;
52:             WHILE pivot<a[j] DO
53:               j:=j-1;
54:             IF i<=j THEN
55:               BEGIN
56:                 help:=a[i];
57:                 a[i]:=a[j];
58:                 a[j]:=help;
59:                 i:=i+1;
60:                 j:=j-1;
61:               END;
62:           UNTIL i>j;
63:         END;
64:       IF l<j THEN
65:         IF j-l>toggle THEN
66:           sort(l,j)
67:         ELSE
68:           direktes_einfuegen(l,j);
69:       IF i<r THEN
70:         IF r-i>toggle THEN
71:           sort(i,r)
72:         ELSE
73:           direktes_einfuegen(i,r);
74:     END;  {sort}
75:
76:   BEGIN {bestsort}
77:     sort(1,feld.max);
78:   END;  {bestsort}
```

```
 1:  { Listing 9d   : 'bestsort' ?? Sortieren mit Quicksort
  fuer lange
 2:                  Teillisten und eingelagertem Sortieren
                          mit direktem
 3:                  Einfuegen fuer kurze Teillisten.
 4:   programmiert : November '87
 5:   von          : Dirk Brockhaus
 6:   mit          : PASCAL+ (CCD)
 7:  }
 8:
 9:  PROCEDURE bestsort;
10:
11:   CONST toggle = 20;
12:
13:   PROCEDURE direktes_einfuegen(l ,
14:                                r : integer);
15:
16:     VAR i    ,
17:         j    : integer;
18:         help : key_type;
19:
20:     BEGIN {direktes_einfuegen}
21:       WITH feld DO
22:         FOR i:=l+1 TO r DO
23:           BEGIN
24:             help:=a[i];
25:             j:=i-1;
26:             WHILE (j>0) AND (a[j]>help) DO
27:               BEGIN
28:                 a[j+1]:=a[j];
29:                 j:=j-1;
30:               END;
31:             a[j+1]:=help;
32:           END;
33:     END;  {direktes_einfuegen}
```

```
 1:  { Listing 1b)
 2:
 3:   Dieses Modul stellt die Operationen auf dem
        abstrakten Datentyp STACK
 4:   zur Verfuegung:
 5:
 6:   Funktionalitaet der Opeartionen:
 7:
 8:     - CREATE: () -> STACK
 9:     - IS_EMPTY: STACK -> BOOLEAN
10:     - PUSH: STACK X STACK_DATA -> STACK
11:     - POP: STACK -> STACK
12:     - TOP: STACK -> STACK_DATA
13:
14:   Reservierte Woerter : CREATE, IS_EMPTY, PUSH, POP und
        TOP
15:
16:   programmiert       : April 87
17:
18:   von                : Dirk Brockhaus
19:  }
20:
21: PROCEDURE CREATE(VAR X:STACK);
22:   BEGIN
23:    X:=NIL;
24:   END;
25:
26: FUNCTION IS_EMPTY(X:STACK):BOOLEAN;
27:   BEGIN
28:    IS_EMPTY:=X=NIL;
29:   END;
30:
31: PROCEDURE PUSH(VAR X:STACK; A:STACK_DATA);
32:   VAR NEW_ELEMENT : STACK;
33:    BEGIN
```

Listing geht weiter...

Fortsetzung Listing 1a

```
34:     NEW(NEW_ELEMENT);
35:     NEW_ELEMENT^.DATA:=A;
36:     NEW_ELEMENT^.NEXT:=X;
37:     X:=NEW_ELEMENT;
38:    END;
39:
40: PROCEDURE POP(VAR X : STACK);
41:  VAR OLD_ELEMENT : STACK;
42:  BEGIN
43:   IF NOT IS_EMPTY(X) THEN
44:     BEGIN
45:      OLD_ELEMENT:=X;
46:      X:=X^.NEXT;
47:      DISPOSE(OLD_ELEMENT);
48:     END;
49:  END;
50:
51: PROCEDURE TOP(X : STACK; VAR A : STACK_DATA);
52:  BEGIN
53:   IF NOT IS_EMPTY(X) THEN
54:     A:=X^.DATA;
55:  END;
```

```
1: { Listing 1a (modifiziert fuer Quicksort iterativ)
2:
3:   Dieses Modul stellt die Typen fuer den abstrakten
        Datentyp STACK zur
4:   Verfuegung.
5:
6:   Reservierte Woerter : stack, stack_element,
        stack_data, data und next.
7:
8:   programmiert : April 87
9:   von          : Dirk Brockhaus
10:  mit          : PASCAL+ (CCD)
11: }
12:
13: stack = ^stack_element;
14:
15: stack_data = RECORD
16:               l ,
17:               r : integer;
18:              END;
19:
20: stack_element = RECORD
21:                  data : stack_data;
22:                  next : stack;
23:                 END;
```

ENDE

Programmier Praxis

Willkommen,

zu unserer letzten Programmierpraxis vor der Sommerpause. Wir haben diesmal ein schnelles Netzgrafikprogramm, das die Bewegung einer Wasseroberfläche darstellt, auf die ein Wassertropfen fällt.

Wer wollte nicht schon immer mal ein eigenes Poster erzeugen und an die Wand hängen. Der PosterMaker macht's möglich.

Anschließend stellen wir ein kleines Programm vor, mit dem es möglich ist, eigene Bilder in eine Resource einzubinden. Denn Bilder verschönern eines Jeden Dialogbox. Doch was tun, wenn das Lieblings-RCS selbige stiefmütterlich behandelt? Ganz einfach: Die Bilder mit einem Malprogramm malen und eine Resource draus machen.

Zuletzt bieten wir noch eine schnelle Textausgabe in Assembler an, die aber auch von BASIC aus genutzt werden kann.

INHALT

GEORG

Sven Geier

Wer sagt eigentlich, daß ein Programm, das dreidimensionale, bewegte Graphik erzeugt, immer aus endlosen Listings bestehen muß? Das hier vorgestellte Programm benötigt genau 53 Zeilen Quelltext, um die Oberfläche eines quadratischen Wasserbeckens darzustellen, in das gerade ein Tropfen gefallen ist. Es berechnet dazu 23 Einzelbilder, die danach fließend hintereinander dargestellt werden.

Das kann ein paar Minuten dauern. Durch bewußt kompaktierende Programmierweise ließen sich dabei sogar noch einige Zeilen einsparen (z.B., indem man die vier DIM-Anweisungen im ersten Teil zu einer zusammenfaßt etc).

Nach etwas Initialisierungskram werden aus der momentanen Bildschirmauflösung zwei Faktoren Fx% und Fy% gewonnen, durch die alle X- bzw. Y-Koordinaten geteilt werden müssen, um der jeweiligen Auflösung gerecht zu werden.

Sodann werden die X-Koordinaten der 6400 (80x80) betrachteten Punkte berechnet. Dies braucht nur einmal zu geschehen, da diese konstant sind. Schließlich kommt der eigentliche Zeichenalgorithmus. Gezeichnet wird ein Cosinus des Abstandes vom Ursprung (in der Mitte des Quadrats), umso stärker gedämpft, je größer dieser Abstand ist. Der Cosinus ist dabei um einen konstanten Phasenwinkel verschoben. Dieser ist n*Pi/11.5, wobei n die Nummer des gerade gezeichneten Bildes ist. Nach 23 Bildern ist der Winkel = 23*Pi/11.5 = 2*Pi, also quasi wieder gleich Null.

An dieser Stelle bieten sich eine Menge Möglichkeiten, das Programm zu variieren. Beispielsweise könnte man die Dämpfung quadratisch zum Abstand verlaufen lassen, indem man statt A=Sqr... schreibt A=1+Sqr... und in der nächsten Zeile das F=(57-A)*Cos... durch F=57/A*Cos... ersetzt. Auch völlig andere Funktionen (sin statt cos etc) können hier eingegeben werden. In diesem Zusammenhang ist es vielleicht wichtig zu wissen, daß die Werte für x1% und y1% im Intervall [-40;40] liegen und daß die Funktionsmaxima im Bereich [~-30;~70] liegen sollten, da sonst die Ausgabe angepaßt werden muß.

Zur Ausgabe ist zu sagen, daß hier ein echter Hidden-Line Algorithmus Verwendung findet, was bedeutet, daß GANZE Linien entweder gezeichnet oder weggelassen werden. Dies ist zwar nicht perfekt, da keine Linien angeschnitten werden, jedoch hält sich der Fehler in Grenzen, da die Linien sowieso nur eine durchschnittliche Länge von 5-6 Pixeln haben. Alles andere wäre (für meine Begriffe) zu langsam gewesen.

Der Algorithmus selbst ist verhältnismäßig einfach: von jedem gezeichneten Anfangspunkt einer Linie wird die Höhe abgespeichert. Liegt ein neu zu zeichnender Punkt unterhalb eines anderen, bereits in dieser Spalte gezeichneten, so ist er unsichtbar (Hierbei nutze ich aus, daß die Ebene von "vorn" nach" hinten" gezeichnet wird). Schließlich werden die 23 so gewonnenen Bilder in rascher Abfolge hintereinander in den Bildschirmspeicher kopiert. Auch hier könnten noch einige Verbesserungen eingebaut werden. Beispielsweise könnte man sich die Kopiererei mit Sput sparen und stattdessen die Anfangsadresse des physikalischen Bildschirmspeichers mit Xbios (5,L:1,L:Varptr (C$(Z%)),-1) auf die abgespeicherten Bildschirme legen oder etwa die Zeiten zwischen den Bildwechseln variabel machen (Zeitlupe etc). Der Phantasie des Programmierers sind also keine Grenzen gesetzt.

Die kompilierte Version brauchte bei mir 10:12 Minuten, also weniger als 25 Sekunden pro Bild. Dies läßt sich wohl auch noch verbessern, indem man beispielsweise die beiden Print-Anweisungen wegläßt, oder für Feine Integervariable verwendet.

Wem es Spaß macht, der kann die Bilder auch gerne vorher abspeichern, um sich bei späteren Läufen des Programms die Rechenzeit zu ersparen. Dies würde allerdings dem Zweck des Programms diametral entgegenlaufen. Ziel war es schließlich, den gewünschten Effekt mit einem möglichst KURZEN Programm zu erreichen. Wenn dieses kurze Programm jedoch erst 750K Daten laden muß, ist der Effekt verschwunden.

PP

```
 1:  Dim C$(22)      ! Um Bilder abzuspeichern
 2:  Cf=Pi/11.5    ! Macht bei 23 Bildern =2*pi=1 Vollkreis
 3:  Hidem         ! Die hält nur auf (u.stört die Graphik)
 4:  R%=Xbios(4)     ! Abfrage der Auflösung
 5:  Fx%=1-(R%=0)    ! Faktor für x-Umwandlung
 6:  Fy%=1-(R%<2)    ! Faktor für y-Umwandlung
 7:  Max_x%=640/Fx% ! Bildschirmbreite
 8:  Max_y%=400/Fy% ! Bildschirmhöhe
 9:  Dim My%(Max_x%)  ! Für Hidden-Line-Algoritmus
10:  Dim X%(80,80)    ! Die X-Koordinaten...
11:  Dim Y%(80,80)    ! ...und die Y-Koordinaten für die
                        6400 (80*80) Punkte
12: '
13:  For X%=0 To 80
14:    For Y%=0 To 80
15:      X%(X%,Y%)=(20+X%*5+Y%*3)/Fx%
16:      If X%(X%,Y%)>Max_x%   ! Berechnung der realen
17:        X%(X%,Y%)=Max_x%   ! X-Koordinaten (konstant!!)
18:      Endif
19:    Next Y%
20:  Next X%
21: '
22:  For C%=0 To 22                 ! Dreiundzwanzig Bilder
23:    Print 22-C%                  ! Ausdruck
24:    Cft=C%*Cf                    ! Phasenwinkel
25:   Arrayfill My%(),Max_y%        ! Für Hidden-Line
26:    For X%=0 To 80               ! Achzig Zeilen
27:      For Y%=0 To 80             ! Achzig Spalten
28:        X1%=X%-40                 !\
29:        Y1%=Y%-40              ! rel. Koordinatensystem
30:        A=Sqr(X1%*X1%+Y1%*Y1%) ! Abstand vom Ursprung...
31:        F=(57-A)*Cos(A/4-Cft)  ! ...dynamische Cosinusfkt.
32:        Y%(X%,Y%)=(350-(Y%*3+F))/Fy%  ! in Bildschirm-
                                          koordinaten.
33:      Next Y%                        ! nächste Spalte
34:      Print At(1,3);80-X%'           ! Information
35:    Next X%                          ! das wars schon
36:  '
37:    Cls
38:    For Y%=0 To 79
39:      For X%=0 To 79
40:        If Y%(X%,Y%)<My%(X%(X%,Y%))   ! Falls sichtbar..
41:          Draw X%(X%,Y%),Y%(X%,Y%) To X%(X%+1,Y%),
                 Y%(X%+1,Y%) To X%(X%+1,Y%+1),Y%(X%+1,Y%+1)
42:          My%(X%(X%,Y%))=Y%(X%,Y%)
43:        Endif
44:      Next X%
45:    Next Y%
46:    Sget C$(C%)
47:  Next C%
48: '
49:  Do
50:    For Z%=0 To 22
51:      Sput C$(Z%)
52:    Next Z%
53:  Loop
```

Listing von Geiers Echt Optimale RasterGrafik

Eines der 23 Einzelbilder

ENDE

PosterMaker druckt riesig

Robert Tolksdorf

Enthusiasten, die sich ihre Computer-Grafiken an die Wand hängen wollen, können jetzt mit dem PosterMaker in neue Dimensionen vorstoßen: Jedes Bild kann bis zu 1,62 mal 1,12 Meter ausgedruckt werden.

Nach dem Programmstart und der Anzeige einer kleinen Info-Box kann mit einer File-Select-Box ein Bild ausgesucht werden.

Das Programm ist in der vorliegenden Form auf Monochrom-Bildschirm eingestellt und kann keine Farbbilder konvertieren. Sie können also normale Bilder im Doodle-Format benutzen.

In zwei weiteren Alertboxen (Bild 1) werden Sie dann nach den Vergrößerungsfaktoren gefragt. PosterMaker kann in jede Richtung 2, 4 oder 8 mal vergrößern, so daß die größte Stufe eine 64-fache Größe bedeutet. Nach einer kleinen Rückfrage (der Ausdruck dauert schon sehr lange) beginnt die Vergrößerung.

Dazu wird das Bild in Abhängigkeit von der gewünschten Vergrößerung in Blöcke unterteilt, die bis auf Bildschirmgröße gebracht werden. PosterMaker schickt dann den Schirm als Hardcopy an den Drucker. Ihnen bleibt nach Ausdruck aller Blöcke noch das Zusammenkleben der Teile und schon haben Sie Ihr Lieblingsbild riesengroß.

PosterMaker druckt jeweils zwei untereinanderliegende Blöcke auf eine Seite und schickt dann ein FormFeed. Somit sparen Sie sich die Hälfte der Klebestellen in vertikaler Richtung. Der Druckvorgang kann übrigens jederzeit durch Drücken einer beliebigen Taste unterbrochen werden. Vor der Ausgabe der gerade bearbeiteten Seite werden Sie gefragt, ob der Ausdruck wirklich abgebrochen werden soll. Ist ein komplettes Bild bearbeitet können Sie durch Auswahl in einer weiteren Alertbox andere Grafiken ausdrucken.

Cut & Paste

Als kleine Hilfe beim Zusammenkleben ist in Bild 2 noch die Ausdruckreihenfolge für einige Vergrößerungsstufen angegeben. Nach dem gleichen Prinzip müssen Sie auch bei den anderen vorgehen.

Das Programm in Modula-2 (Listing 1) macht ausgiebig Gebrauch von Rastercopies. Die Einzelheiten zu dieser Funktion wurden in der ST-Ecke in ST-Computer 11/87, Seite 76 erläutert. Hier wird sie auf zweierlei Weise verwendet.

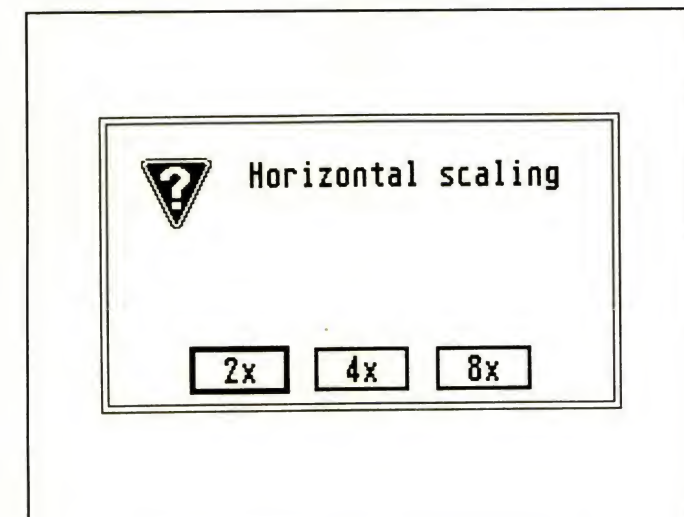
Die Prozedur "SetUpPicScreen" reserviert 32000 Bytes im Speicher. Dieser Bereich kann vom VDI wie ein realer Bildschirm verwendet werden, wenn ein entsprechender Memory-Form-Definition-Block definiert wurde. Das auszudruckende Bild wird nun in diesen "Speicherbildschirm" geladen und jeweils mit einer Rastercopy blockweise auf den realen kopiert, was in der Prozedur "Blow ItUp" geschieht.

Aber auch zum Vergrößern selbst läßt sich die Rastercopy-Funktion einsetzen und sichert dabei eine schnelle Verarbeitung. Die Vorgehensweise dabei ist einfach. Bei der Vergrößerung in vertikaler Richtung auf das Doppelte geht die untere Bildschirmhälfte sowieso verloren. Wir nehmen also die letzte Bildschirmzeile der ersten Hälfte und kopieren sie einfach in die unterste Bildschirmzeile. Wird ihr Inhalt auch noch in die vorletzte Zeile kopiert, ist eine Pixelreihe vergrößert. Auf diese Art arbeitet sich die Prozedur "BlowUpV" bis zur ersten Bildschirmzeile hoch.

Bei der horizontalen Vergrößerung mit "BlowUpH" werden einfach alle Bildschirmspalten ab der Mitte bis an den linken Rand je zweimal koopiert. Eine kleine Erläuterung verdient noch die Funktion "Power Of2", die mit dem Parameter "a" zwei hoch a liefert. Der Trick dabei ist, daß die Standardroutine "INCL" benutzt wird. Sie setzt das Bit Nummer a in dem Bitvektor b. Wer sich im binären Zahlensystem auskennt, weiß sofort, daß dieser Bitvektor nun dem Wert zwei hoch a entspricht, und genau das liefert die Funktion als CARDINAL ab.

Bleibt nur noch zu hoffen, daß Sie für den PosterMaker genügend Farbbänder auf Vorrat haben und das Sägen Ihres Druckers auch auf längere Zeit ertragen können!





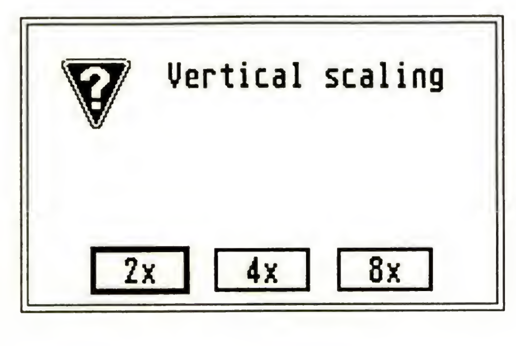


Bild 1: Alertboxen um die Vergrößerungsfaktoren zu bestimmen

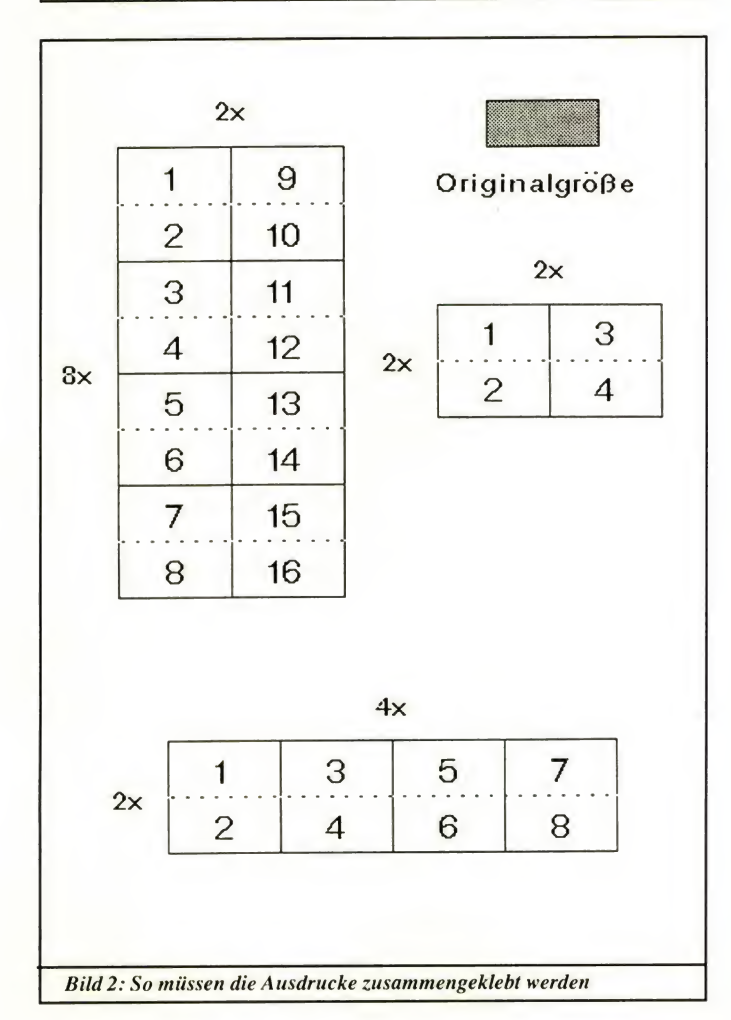


Bild 2: So müssen die Ausdrucke zusammengeklebt werden

```
 1: MODULE PosterMaker;
 2:
 3: (* Robert Tolksdorf, 9.12.87       *)
 4:
 5: (*$A+*)          (* Code optimieren *)
 6:
 7: FROM SYSTEM             IMPORT ADR, ADDRESS ;
 8: FROM ASCII              IMPORT FF ;
 9: FROM Strings            IMPORT String, Length, Concat,
                            InitStringModule ;
10: FROM GEMDOS             IMPORT Open, Read, Close, ConIS,
                                   NecIn, PrnOut,
11:                                Alloc, Free ;
12: FROM GEMAESbase         IMPORT Arrow, MouseOff, MouseOn;
13: FROM AESApplications    IMPORT ApplInitialise, ApplExit;
14: FROM AESForms           IMPORT FormAlert,
                                   FileSelectorInput ;
15: FROM AESGraphics        IMPORT GrafMouse ;
16: FROM GEMVDIbase         IMPORT VDIWorkInType,
                                   VDIWorkOutType,
17:                                BigPxyArrayType ;
18: FROM VDIControls        IMPORT OpenVirtualWorkstation,
19:                                CloseVirtualWorkstation;
20: FROM VDIRasters         IMPORT MFDBType,
                                   CopyRasterOpaque ;
21: FROM XBIOS              IMPORT ScreenPhysicalBase,
                                   ScreenDump ;
22:
23: CONST
24:   InfoAlert   = '[0][ The Poster Maker|  from ST-
    Computer|written Dez. 1987 by|  Robert Tolksdorf| ]
      [ OK ]';
25:   HorizScale  = '[2][Horizontal scaling]
                     [ 2x | 4x | 8x ]';
26:   VertScale   = '[2][Vertical scaling][ 2x | 4x | 8x ]';
27:   ReallyPrint = '[3][Really start printing ?]
                     [ OK | Cancel ]';
28:   AbortPrint  = '[2][Really abort printing ?]
                     [ OK | Cancel ]';
29:   Another     = '[2][Another printout ?][ Yes | No ]';
30:
31:   DEBUG       = FALSE; (* TRUE, wenn kein Printout *)
32:
33: VAR PicScreen, RealScreen   : MFDBType;
      (* Screen Informationen *)
34:     PicScreenA, RealScreenA : ADDRESS;
      (* deren Adressen        *)
35:     ApplID, ScreenHandle    : INTEGER;
      (* AES/VDI ID's          *)
36:     ScreenSize              : LONGCARD;
      (* BildschirmgröPe       *)
37:     Scaling                 : RECORD
      (* VergröPerung          *)
38:                                 Horiz,
39:                                 Vert  : CARDINAL;
40:                               END;
41:
42: (* Screen-Init, MFDB's setzen,
      Speicherbildschirm einrichten    *)
43: PROCEDURE SetUpPicScreen;
44: VAR workin   : VDIWorkInType;
45:     workout  : VDIWorkOutType;
46:     i        : INTEGER;
47: BEGIN
48:   InitStringModule;
49:   i:=FormAlert(1,InfoAlert);
50:   FOR i:=0 TO 9 DO workin[i]:=0 END;
51:   workin[10]:=2;
52:   OpenVirtualWorkstation(workin,ScreenHandle,workout);
53:   WITH RealScreen DO
54:     pointer:=ScreenPhysicalBase();
55:     width:=639;
56:     height:=399;
57:     widthW:=40;
58:     format:=1;
59:     planes:=1;
60:   END;
61:   RealScreenA:=ADR(RealScreen);
62:   PicScreen:=RealScreen;
63:   Alloc(ScreenSize,PicScreen.pointer);
64:   PicScreenA:=ADR(PicScreen);
65: END SetUpPicScreen;
66:
67: (* File auswählen und eventuell in Speicherbildschirm
      laden      *)
```

Listing geht weiter...

```
 68: PROCEDURE LoadPic():BOOLEAN;
 69:  VAR inpath, insel, filename : String ;
 70:      dummy                   : BOOLEAN;
 71:      exbutton, handle        : INTEGER;
 72:      status                  : (StillSelecting, Aborted,
                                     ReadIn);
 73:  BEGIN
 74:   GrafMouse(Arrow, NIL);
 75:   status:=StillSelecting;
 76:    REPEAT
 77:     inpath:='\*.*';
 78:     insel[0]:=0C;
 79:   FileSelectorInput(ADR(inpath),ADR(insel),exbutton);
 80:     IF (exbutton#0) AND (insel[0]#0C) THEN
 81:       WHILE inpath[Length(inpath)-1]#'\' DO
 82:         inpath[Length(inpath)-1]:=0C
 83:       END;
 84:      Concat(inpath,insel,filename);
 85:      Open(filename,0,handle);
 86:       IF handle>0 THEN
 87:       Read(handle,ScreenSize,PicScreen.pointer);
 88:         dummy:=Close(handle);
 89:         status:=ReadIn;
 90:       END;
 91:     ELSE
 92:       status:=Aborted;
 93:     END;
 94:   UNTIL (status#StillSelecting);
 95:   RETURN (status=ReadIn);
 96: END LoadPic;
 97:
 98: (* Bild anzeigen u.mit Alerts VergröPerung auswählen *)
 99: PROCEDURE SelectScaling():BOOLEAN;
100:  VAR pxy : BigPxyArrayType;
101:  BEGIN
102:    pxy[0]:=0;
103:   pxy[2]:=PicScreen.width;
104:    pxy[4]:=0;
105:   pxy[6]:=RealScreen.width;
106:    pxy[1]:=0;
107:   pxy[3]:=PicScreen.height;
108:    pxy[5]:=0;
109:   pxy[7]:=RealScreen.height;
110:   GrafMouse(MouseOff,NIL);
111:    (* Auf den realen Bildschirm kopieren *)
112:  CopyRasterOpaque(ScreenHandle,3,pxy,PicScreenA,
                       RealScreenA);
113:   GrafMouse(MouseOn,NIL);
114:  Scaling.Horiz:=CARDINAL(FormAlert(1,HorizScale));
115:  Scaling.Vert :=CARDINAL(FormAlert(1,VertScale));
116:   RETURN (FormAlert(1,ReallyPrint)=1);
117: END SelectScaling;
118:
119: (* Blockweise in den realen Bildschirm kopieren,
     vergröPern und ausdrucken.
120:    Bei Tastendruck eventuell abbrechen *)
121: PROCEDURE BlowItUp;
122: VAR i, hblock, vblock, hblocksize, vblocksize : INTEGER;
123:     pxy                                : BigPxyArrayType ;
124:     ch                                 : CHAR;
125:
126:   (* 2 hoch a ausrechnen *)
127:   PROCEDURE PowerOf2(a:CARDINAL):CARDINAL;
128:    VAR b:BITSET;
129:    BEGIN
130:      b:={};          (* Trick mit BITSET: Das Bit # a  *)
131:      INCL(b,a);      (* wird gesetzt = 2 hoch a *)
132:     RETURN CARDINAL(b); (* und als CARDINAL
                              zurückgegeben *)
133:   END PowerOf2;
134:
135: BEGIN
136:  hblocksize:=(PicScreen.width+1) DIV
                         PowerOf2(Scaling.Horiz);
137:  vblocksize:=(PicScreen.height+1) DIV
                         PowerOf2(Scaling.Vert);
138:    pxy[4]:=0;  (* Destination-x und          *)
139:   pxy[5]:=0;  (* Destination-y sind immer 0 *)
140:  GrafMouse(MouseOff,NIL);
141:    (* blockweise ...       *)
142:  FOR hblock:=1 TO PowerOf2(Scaling.Horiz) DO
143:    FOR vblock:=1 TO PowerOf2(Scaling.Vert) DO
144:      (* ... realen Bildschirm löschen ... *)
145:       pxy[0]:=0;
146:       pxy[2]:=RealScreen.width;
147:       pxy[6]:=RealScreen.width;
148:        pxy[1]:=0;
149:       pxy[3]:=RealScreen.height;
150:       pxy[7]:=RealScreen.height;
151:     CopyRasterOpaque(ScreenHandle,0,pxy,RealScreenA,
                          RealScreenA);
152:       (* ... Block aus Speicher kopieren ... *)
153:      pxy[0]:=(hblock-1)*hblocksize;
154:      pxy[2]:=hblock*hblocksize-1;
155:       pxy[6]:=hblocksize-1;
156:      pxy[1]:=(vblock-1)*vblocksize;
157:      pxy[3]:=vblock*vblocksize-1;
158:       pxy[7]:=vblocksize-1;
159:     CopyRasterOpaque(ScreenHandle,3,pxy,PicScreenA,
                          RealScreenA);
160:       (* ... vergröPern ... *)
161:      FOR i:=1 TO Scaling.Horiz DO
162:        BlowUpH;
163:      END;
164:      FOR i:=1 TO Scaling.Vert DO
165:        BlowUpV;
166:      END;
167:      (* ... Taste gedrückt ?? *)
168:      IF ConIS() THEN
169:        NecIn(ch);
170:       GrafMouse(MouseOn,NIL);
171:       IF (FormAlert(2,AbortPrint)=1) THEN
172:         RETURN  (* Abbruch *)
173:        ELSE
174:         GrafMouse(MouseOff,NIL);
175:        END
176:      END;
177:      IF ~DEBUG THEN
178:        (* realen Bildschirm an Drucker schicken *)
179:        ScreenDump ;
180:         (* Seite voll -> FormFeed              *)
181:        IF ~ODD(vblock) THEN
182:          PrnOut(FF);
183:        END;
184:      END;
185:    END;
186:  END;
187:  GrafMouse(MouseOn,NIL);
188: END BlowItUp;
189:
190: (* realen Bildschirm horiz. a.d.Doppelte vergröPern *)
191: PROCEDURE BlowUpH;
192:  VAR x   : INTEGER;
193:      pxy : BigPxyArrayType;
194:  BEGIN
195:   pxy[1]:=0;
196:  pxy[3]:=RealScreen.height;
197:   pxy[5]:=0;
198:  pxy[7]:=RealScreen.height;
199:   FOR x:=((RealScreen.width+1) DIV 2)-1 TO 0 BY -1 DO
200:     pxy[0]:=x;
201:     pxy[2]:=x;
202:     pxy[4]:=x*2;
203:     pxy[6]:=x*2;
204:   CopyRasterOpaque(ScreenHandle,3,pxy,RealScreenA,
                        RealScreenA);
205:     INC(pxy[4]);
206:     INC(pxy[6]);
207:   CopyRasterOpaque(ScreenHandle,3,pxy,RealScreenA,
                        RealScreenA);
208:   END;
209: END BlowUpH;
210:
211: (* realen Bildschirm vertikal a.d.Doppelte vergröPern *)
212: PROCEDURE BlowUpV;
213:  VAR y   : INTEGER;
214:      pxy : BigPxyArrayType;
215:  BEGIN
216:   pxy[0]:=0;
217:  pxy[2]:=RealScreen.width;
218:   pxy[4]:=0;
219:  pxy[6]:=RealScreen.width;
220:   FOR y:=((RealScreen.height+1) DIV 2)-1 TO 0 BY -1 DO
221:     pxy[1]:=y;
222:     pxy[3]:=y;
223:     pxy[5]:=y*2;
224:     pxy[7]:=y*2;
225:   CopyRasterOpaque(ScreenHandle,3,pxy,RealScreenA,
                        RealScreenA);
```

Listing geht weiter...

```
226:      INC(pxy[5]);
227:      INC(pxy[7]);
228:    CopyRasterOpaque(ScreenHandle,3,pxy,RealScreenA,
                         RealScreenA);
229:    END;
230:  END BlowUpV;
231:
232:  (* Speicher freigeben und Screen schliePen *)
233: PROCEDURE CloseDown;
234: VAR dummy:BOOLEAN;
235:  BEGIN
236:   dummy:=Free(PicScreen.pointer);
237:   CloseVirtualWorkstation(ScreenHandle);
238:  END CloseDown;
239:
240: BEGIN
241:   ApplID:=ApplInitialise();
242:   ScreenSize:=32000;
243:   SetUpPicScreen;
244:    REPEAT
245:      IF (LoadPic()) AND (SelectScaling()) THEN
246:         BlowItUp;
247:       END;
248:   UNTIL (FormAlert(1,Another)=2);
249:    CloseDown;
250:    ApplExit;
251: END PosterMaker.
```

Listing vom PosterMaker

ENDE

Die "Programmier Praxis" ist ein fester Bestandteil der ST Computer geworden. Gedacht ist sie für all diejenigen, die ihren Rechner selbst programmieren, und dazu Tips und Anregugen gebrauchen können. Diese Rubrik kann aber nur dann bestehen, solange viele Leser (Sie eingeschlossen) sich daran beteiligen. Wir forden Sie deshalb auf Ihren Ideen in einen Umschlag zu stecken und auf dem schnellsten Weg zu uns zu schicken. Voraussetzungen gibt es praktisch keine: es muß lediglich interessant sein und so dokumentiert, daß auch andere Anwender daraus einen Nutzen ziehen können. Auch bei den Sprachen gibt es keinerlei Einschränkungen: PASCAL, C, BASIC, Assembler, Modula 2, APL, FORTRAN, LISP, Prolog, PEARL,.... und sogar LOGO ist erwünscht.
Veröfentliche Programme werden natürlich angemessen honoriert. Bitte beachten Sie bei der Einsendung: Schicken Sie den Quelltext und das evtl.compilierte Programm, sowie die Dokumentacion auf Papier und Diskette. Die Diskette wird zurückerstattet.
Für weiterer Informationen lesen Sie bitte die zwei Auskunftsseiten "Möchten Sie bei Uns schreiben?" in diesem Heft.

Kontaktadresse:
´Merlin`-Computer GmbH
ST-COMPUTER Redaktion
Z.Hd. Herrn Merino
Kennwort: Programmier Praxis
Industriestr. 26
6236 Eschborn

Eigene Bilder in der Resource

Ingo Brümmer

Bilder verschönern eines Jeden Dialoggbox. Doch was tun, wenn das Lieblings-RCS selbige stiefmütterlich behandelt? Ganz einfach: Die Bilder mit einem Malprogramm malen und eine Resource draus machen. RCS) von Digital Research kann Bilder (Bitblk-Strukturen) nur hinzuladen und erwartet dabei einen C-Quellcode. Das RCS von Kuma besitzt zwar einen eigenen Icon-Editor, doch oft genug hat man das Bild schon und will es nicht noch einmal abmalen.

Außerdem soll es ja auch Leute geben, die mit der Lupe ihres Malprogramms besser zurechtkommen als mit einem Icon-Editor. So kam mir die Idee, ein Programm zu schreiben, daß einen beliebigen Bildschirmausschnitt in eine Bitblk-Struktur übersetzt. Besitzer des Kuma-RCS können so auch den Icon-Editor entlasten, denn ihr RCS ermöglicht es bequem, Bitblk-Strukturen in Icon-Strukturen zu wandeln. Sie können sich also fortan die Bilder für Ihre Resource mit Ihrem Lieblings- Malprogramm malen. Das GFA-Programm, das die Konvertierung vornimmt, finden Sie im nebenstehenden Listing.

Zur Vorgehensweise des Programms: Zunächst wählen Sie ein Bild aus, aus dem Sie den Ausschnitt haben wollen. Dieses muß im Screenformat (32kB) vorliegen. Dann können Sie mit einer 'Gummibandbox' den gewünschten Ausschnitt markieren. Dieser wird im String 'Aus$' gespeichert und dann in die Bitblk-Struktur umgewandelt. Diese wird in eine komplette Resource eingelagert, die mit jedem RCS zu laden ist. Zur Erzeugung der Bitblk-Struktur müssen allerdings zunächst Höhe und Breite des Ausschnitts bekannt sein. Jeder String, in dem per 'GET' ein Bildschirmteil gespeichert wurde, besitzt einen sechs Bytes langen Vorspann, in dem im ersten Wort die Breite und im zweiten die Höhe (jeweils in Pixeln) des Ausschnitts vermerkt sind. Das dritte Wort enthält Informationen über die Bildschirmauflösung, Farbebenen etc.

Jede neue Pixelzeile des Ausschnitts beginnt an einer Wortadresse. Das bedeutet, daß bei einem 15 Pixel breiten Ausschnitt also jeweils das letzte Bit der Wörter 'verschenkt' wird. Wäre dieser Ausschnitt wiederum 15 Pixel hoch, so ergäbe sich für die Länge des Strings in dem er gespeichert ist:

6+15*2 = 36 Bytes

Der String wäre wegen der Wortlänge jeder Zeile genauso lang, wenn der Ausschnitt nur ein Pixel breit wäre. Leider gibt es bei diesen Werten, die man aus dem Vorspann erfahren kann, einige Ungereimtheiten. Ein 16 Pixel breiter Ausschnitt nimmt nicht - wie vermutet - pro Zeile zwei Byte in Anspruch, sondern vier. Also wird bei jedem 16. Pixel bereits ein neues Wort begonnen. Ein Ausschnitt, für den im Vorspann eine Höhe von Null Pixeln vermerkt ist, ist aber in Wirklichkeit ein Pixel hoch. Daraus entstehen die zunächst unlogisch wirkenden Berechnungen im Programm. In der Bitblk-Struktur speichert (und erwartet, logo!) das AES folgende Informationen (*siehe Tabelle*) womit wir auf eine Länge von 14 Bytes kommen. Hinzu kommen die Bilddaten: pro angefangenem Wort 2 Bytes oder viel einfacher, Länge von 'Aus$'-6.

Nun macht ja ein Bitblk noch keine Resource. Wir brauchen noch einen sog. Resource-Header, der wiederum 36 Bytes lang wird (Genaueres siehe Listing).

Dann fehlt noch die Objekt-Struktur (24 Bytes) und die Objektbaumtabelle (bei einem Baum 4 Bytes). Das wär's eigentlich schon, doch kann das RCS von Digital Research mit der Resource, die jetzt entstünde, herzlich wenig anfangen. Das liegt daran, daß der Bitblk in dieser Resource eine Vaterfigur spielte, also das Wurzelobjekt seines Baumes darstellte.

Leider erlaubt eben dieses RCS nicht, ein solches Objekt auf das Klemmbrett zu kopieren, um es von dort aus weiter zu verarbeiten. Wir kommen also nicht umhin, dem Bild(chen) ein Vaterobjekt überzuordnen. Aber das soll ja kein Problem sein. Ich habe eine geringfügig größere Box gewählt, in der sich das Bild dann befinden wird. Allerdings erfordert das eine weitere Objektstruktur, mithin

wird die Resource nochmal 24 Bytes länger. Die Länge ist wichtig für den Eintrag 'Rssize' im Resource-Header (das ist der letzte). Haben Sie mitgezählt? Für die Länge ergibt sich:

36 Bytes	*(Header)*
+ Len(Aus$)-6 Bytes	*(Bilddaten)*
+ 14 Bytes	*(Bitblk)*
*+ 2*24 Bytes*	*(Objekte)*
+ 4 Bytes	*(Objektbaumtabelle)*

Es wird nun ein String dieser Länge vorbereitet und von seiner Adresse an werden die Daten per POKE, DPOKE und LPOKE gesetzt. Anschließend wird dieser String unter dem von Ihnen gewählten Dateinamen gespeichert.

Eine kleine Bemerkung vielleicht noch zu den letzten vier Werten der Objektstruktur. Hier sind X- und Y-Position (relativ zum Vaterobjekt) sowie Breite und Höhe der Objekte vermerkt. Während in einer Resource, die mit der AES-Funktion 110 (RSRC_LOAD) geladen wurde, diese Werte in Pixelkoordinaten vermerkt sind, werden sie in der Resource- Datei in Zeichenkoordinaten verlangt. Auf dem Monochrommonitor ist das ein Raster von 16 Pixeln Höhe und 8 Pixeln Breite. Der entsprechend geteilte Wert wird im Low-Byte der jeweiligen Koordinaten-Wörter gespeichert. Sollte die Koordinate nun nicht exakt auf diesem Raster liegen, also ein Rest bleiben, so wird dieser (nun in Pixeln) im High-Byte vermerkt. Beim Laden werden die Werte entsprechend der Bildschirmauflösung umgerechnet. Wenn Sie also das Programm in der mittleren Auflösung verwenden wollen, müssen Sie in der Zeile, in der die Koordinate durch 16 geteilt wird, diese 16 durch eine 8 ersetzen.

Ich hoffe, daß alle anderen Fragen durch das Listing selbst zu beantworten sind. Als dann wünsche ich Ihnen viel Spaß beim Malen!

bi_pdata (Zeiger auf Bilddaten)	*4 Bytes*
bi_wb (Breite in Bytes)	*2 Bytes*
bi_hl (Höhe in Pixeln)	*2 Bytes*
bi_x (X-Position)	*2 Bytes*
bi_y (y-Position)	*2 Bytes*
bi_color (Farbe des Bildes)	*2 Bytes*

Tabelle der Bild-Struktur

```
 1: '*******************************************************
 2:  ' * PIC_BITB.BAS - aus (Bildschirm-)Bildern Bitblöcke
        erstellen                                          *
 3:  ' * Copyright 1988 Ingo Brümmer, LutherstraPe 60,
        3000 Hannover 1                                    *
 4: '*******************************************************
 5:   '
 6:  Repeat
 7:    Print " Bitte eine Bilddatei wählen: "
 8:  Fileselect Chr$(Gemdos(25)+65)+"\*.*","",Bild$
 9:     '
10:     If Bild$=""                    ! Bei Abbruch
11:       Edit                         ! Ende
12:     Endif
13:     Bload Bild$,Xbios(2)           ! Bild laden
14:     '
15:     Repeat
16:       Aus$=""
17:       Defmouse 5                   ! Maus -> Kreuz
18:       Sget Moni$                   ! Bildschirm merken
19:       Repeat
20:       Until Mousek=1               ! Warten auf linke
              Taste
21:       X1%=Mousex                   ! Koordinaten merk
             en
22:       Y1%=Mousey
23:       Do
24:         Mouse X2%,Y2%,T%          ! Maus merken
25:         Box X1%,Y1%,X2%,Y2%       ! Gummibox zeichnen
26:          Vsync                    ! Mit Vertical-Interrupt
                                        synchronisieren
27:          Sput Moni$               ! altes Bild drauf
28:         Exit If T%=0       ! Raus, wenn Maus losgelassen
29:       Loop
30:      Get X1%,Y1%,X2%,Y2%,Aus$   ! Bildchen ausschneiden
31:       Defmouse 0
32:        '
```

```
33:       Put X1%,Y1%,Aus$,12        ! Und invertiert
                                       drauflegen
34:       Pause 25
35:     Alert 0,"Bitblk draus machen?",1,"Ja|Nein",D%
36:       If D%=1
37:       Fileselect Chr$(Gemdos(25)+65)+"\*.*","",Rs$
38:       Endif
39:      If D%=1 And Rs$<>""     ! Wenn Speichern und nicht
                                   Abbruch:
40:         '
41:        B%=Dpeek(Varptr(Aus$))  ! Breite in Pixeln
42:         B%=(B% Div 16+1)*16    ! auf Wortlänge bringen
43:        H%=Dpeek(Varptr(Aus$)+2)+1    ! Höhe in Pixeln
44:         '
45:       Imd$=Right$(Aus$,Len(Aus$)-6)  ! Bilddaten
46:         '
47:       Rsc$=Space$(36+Len(Imd$)+14+2*24+4) ! Platz für
                                              RSC schaffen
48:        Adr%=Varptr(Rsc$)
49:        ' *************** RSHDR basteln ***************
50:         Dpoke Adr%+0,1                ! Vrsn, immer(?)=1
51:        Dpoke Adr%+2,36+Len(Imd$)+14  ! Offset Objekt-
                                              Struktur
52:        Dpoke Adr%+4,36+Len(Imd$)+14  ! Offset Tedinfo-
                                              Struktur
53:        Dpoke Adr%+6,36+Len(Imd$)+14  ! Offset Iconblk-
                                              Strukur
54:         Dpoke Adr%+8,36+Len(Imd$)     ! Offset Bitblk-
                                              Struktur
55:         Dpoke Adr%+10,36+Len(Imd$)    ! Offset freie
                                              Strings
56:          Dpoke Adr%+12,36             ! Offset gebundene
                                              Strings
57:          Dpoke Adr%+14,36             ! Offset Image-
                                              Daten
58:       Dpoke Adr%+16,36+Len(Imd$)+14  ! Offset freie
                                              Images
```

Listing geht weiter…

```
59:   Dpoke Adr%+18,36+Len(Imd$)+14+2*24 ! Offset
                                          Baumtabelle
60:   Dpoke Adr%+20,2        ! Anzahl der Objekte
61:   Dpoke Adr%+22,1        ! Anzahl der Bäume
62:   Dpoke Adr%+24,0        ! Anzahl der Tedinfos
63:   Dpoke Adr%+26,0        ! Anzahl der Iconblöcke
64:   Dpoke Adr%+28,1        ! Anzahl der Bitblöcke
65:   Dpoke Adr%+30,0        ! Anzahl der Strings
66:   Dpoke Adr%+32,0        ! Anzahl der Images
67:   Dpoke Adr%+34,Len(Rsc$)  ! Länge der Resource
68:   '
69:   ' **************** Bilddaten eintragen *********
70:   Bmove Varptr(Imd$),Adr%+36,Len(Imd$)
71:   '
72:   ' ************* Bit Image Block basteln ********
73:   Add Adr%,36+Len(Imd$)   ! Anfangsadresse d.Bitblk
74:   Lpoke Adr%,36          ! Adresse der Bilddaten
75:   Dpoke Adr%+4,B% Div 8  ! Breite in Bytes
76:   Dpoke Adr%+6,H%        ! Höhe in Pixeln
77:   Dpoke Adr%+8,0         ! X-Position
78:   Dpoke Adr%+10,0        ! Y-Position
79:   Dpoke Adr%+12,1        ! Farbe
80:   '
81:   Add Adr%,14  ! Anfangsadresse der Objektstruktur
82:   '
83:   ' ********* Objektstruktur der Vater-Box *******
84:   Dpoke Adr%,-1   ! Nächstes Objekt, hier: keines
85:   Dpoke Adr%+2,1  ! Erstes Kind, hier: Nr.1
86:   Dpoke Adr%+4,1  ! Letztes Kind, hier: Nr.1
87:   Dpoke Adr%+6,20  ! Objekttyp - G_box
88:   Dpoke Adr%+8,0   ! Objektflag
89:   Dpoke Adr%+10,16 ! Objektstatus - Outlined
90:   Lpoke Adr%+12,135424 ! Obj_spec - Langwort
91:   Dpoke Adr%+16,0      ! X-Position
92:   Dpoke Adr%+18,0      ! Y-Position
93:   Dpoke Adr%+20,B% Div 8+1 ! Breite in Spalten
94:   Poke Adr%+22,0        ! (Rest-)Höhe in Pixeln
                              im Highbyte
95:   Poke Adr%+23,H% Div 16+1 ! Höhe in Zeilen im
                                 Lowbyte
96:   Add Adr%,24
97:   '
98:   ' ********* Objektstruktur des Bitblk **********
99:   Dpoke Adr%,0     ! Nächstes Objekt, hier: Vater
100:  Dpoke Adr%+2,-1  ! Erstes Kind, hier: keines
101:  Dpoke Adr%+4,-1  ! Letztes Kind, hier: keines
102:  Dpoke Adr%+6,23  ! Objekttyp - Bitblk
103:  Dpoke Adr%+8,32  ! Objektflag = Lastob
104:  Dpoke Adr%+10,0  ! Objektstatus
105:  Lpoke Adr%+12,36+Len(Imd$) ! Zeiger auf Bitblk -
                                  Langwort
106:  Dpoke Adr%+16,0          ! X-Position
107:  Dpoke Adr%+18,0          ! Y-Position
108:  Dpoke Adr%+20,B% Div 8   ! Breite in Spalten
109:  Poke Adr%+22,H% Mod 16  ! (Rest-)Höhe in Pixeln
                               im Highbyte
110:  Poke Adr%+23,H% Div 16  ! Höhe in Zeilen im
                               Lowbyte
111:  '
112:  ' ************ Objektindex - Tabelle ************
113:  Lpoke Adr%+24,36+Len(Imd$)+14  ! Objektanfang des
                                       0.Baumes
114:  '
115:  Bsave Rs$,Varptr(Rsc$),Len(Rsc$) ! Speichern
116:  Endif
117:  Put X1%,Y1%,Aus$                 ! Invertierung weg
118:  Alert 0," Nochmal oder neues Bild? ",1,
                           "Nochmal|Neu|Ende",D%
119:  Until D%>1
120: Until D%=3
```

Mit diesem Listing können sie ihr Resource einbinden

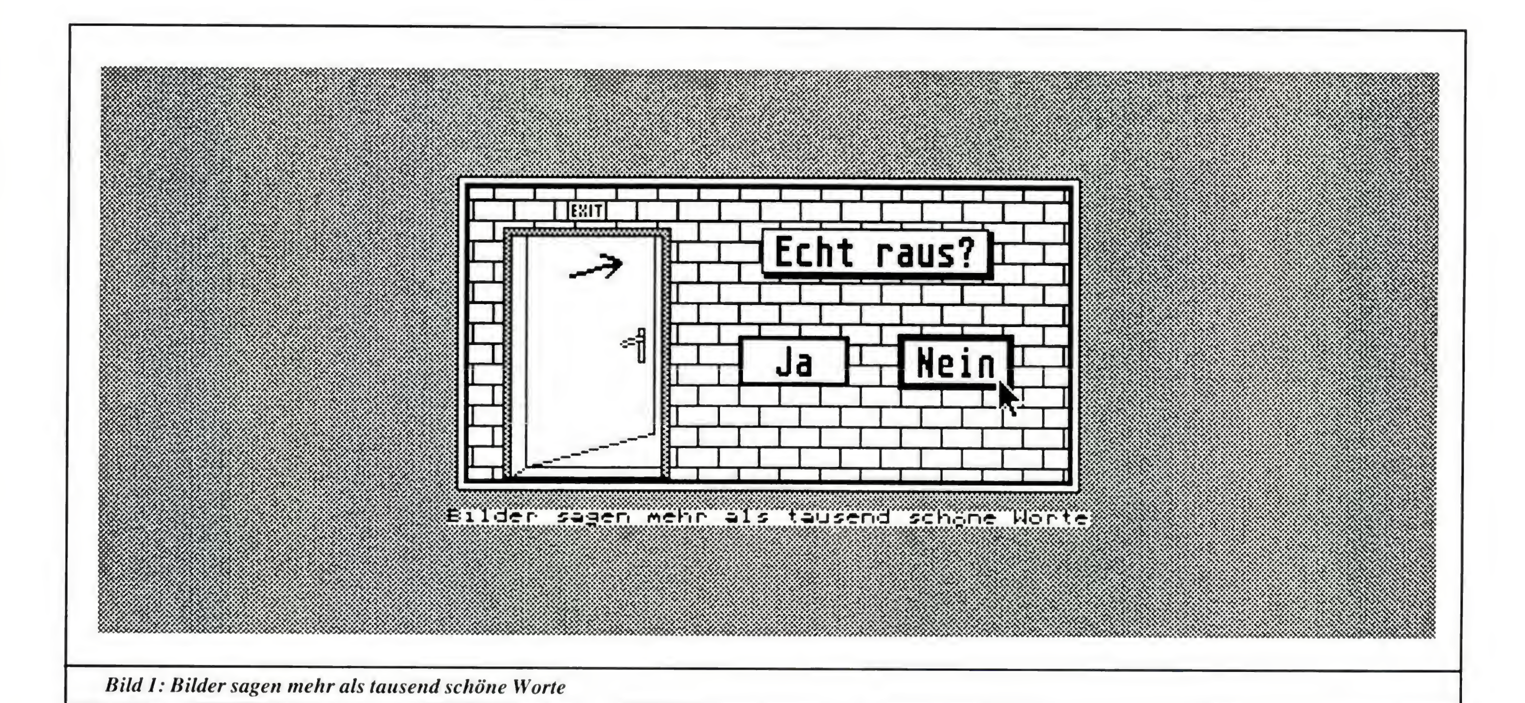


Bild 1: Bilder sagen mehr als tausend schöne Worte

ENDE

Qualität ist meßbar

G SCANNER

Problemloses Digitalisieren von s/w-Bildern mit Ihrem Drucker ermöglicht der G SCANNER. Die hochwertige Abtastdiode von HP garantiert die ausgezeichnete Qualität von 200 dpi. Sie wird einfach auf dem Druckknopf Ihres Druckers befestigt. Mitgeliefert wird Software, die DEGAS und Normalbilder erstellt und Bildeditieren ermöglicht. Der G SCANNER ist voll kompatibel zu STAD und SYMPATIC PAINT (Scannoption).

DM 298,—

P6 Halterung für Diode DM 35,—

Neu: GSCANNER für Graustufen verwandelt Farbbilder in Graustufen. Kompatibel zu GSCANNER und STAD DM 398,—

RELAS

Aktienanalyse für den privaten und professionellen Aktienanleger auf der Basis der **Relativen Stärke und Volatilität** von Aktienkursen — **Charts** wahlweise über Bildschirm und Drucker — **Mischen von Charts** einer Aktie und Branche zum Vergleich — **Kurseingabe nur einmal wöchentlich** (!) Als Basis dienen die letzten 27 Wochenschlußkurse — Umfangreiche Bereinigungsfunktion bei Kapitalveränderungen von Aktien — **Listen** der Relativen Stärke mit Mittelwert von Kursen und Analyse der Veränderung der Rangfolge der Relativen Stärke über 3 Wochen (wahlweise Bildschirm oder Drucker) — u.v.m.

DM 398,—

Handbuch vorab: DM 40,—

ANTI VIREN KIT

Unser neues Anti Viren Kit: erkennt alle bekannten Viren, erkennt **ähnliche** Viren — vernichtet allebekannten Viren und deren Verwandten — erkennt bei vielen kommerziellen Programmen, ob gesund oder infiziert. — Datenrestbestände werden gerettet — **immunisiert** Disketten gegen Bootsektorviren. Die Disketten werden geimpft — repariert defekte und infizierte Bootsektoren — Link Viren werden von den infizierten Programmen abgeschnitten und vernichtet — intelligente Bootsektorscannung; erkennt auch noch unbekannte Viren — **Updateservice und Datenrettungsservice** für registrierte Benutzer.

Für alle ST mit Monochrom- und Farbmonitor DM 99,—

16 BIT: AS SOUND SAMPLER III

Schon der Sound Sampler II überzeugte Kunden wie Presse (Keyboards 12 / 87: »Verglichen mit 10 – 15-fach teureren Sampler… muß man von einer guten bis sehr guten Qualität sprechen.« ST 1 / 8: »Der Alleskönner.«). Alle Leistungsmerkmale des Sampler III bietet der neue Sound Sampler III in 16 Bit CD Qualität! Aufnahmen, editieren, verknüpfen, mischen, transponieren, Echoeffekte, einbinden in Programme, Sounds oder MIDI Keyboards spielen; alles in original CD Qualität! Jetzt in 3 Versionen.

AS Sound Sampler II Standard:	DM 198,—
AS SOUNDSAMPLER II maxi:	DM 298,—
AS SOUND SAMPLER III 16 BIT:	DM 598,—
Soundbibliothek 8 BIT:	DM 149,—
Soundbibliothek 16 BIT:	DM 198,—

G RAMDISK II +

In der neuesten Version mit allen Leistungsmerkmalen, die man sich für eine Ramdisk überhaupt wünschen kann: Resetfest bis 4 MB, dabei abschaltbar; integrierter Druckerspooler, Bootcopy; Parameter save usw. Die G RAMDISK II ist im harten Einsatz bereits tausendfach bewährt.

DM 49,—

CHESS

Das stärkste Schachprogramm für ST Computer (bei 66.000 Eröffnungszügen) bietet nicht nur Spielstärke, sondern auch Komfort. Nachladbare und erweiterbare Eröffnungsbibliothek, beliebige Bedenkzeit (bei Ausnutzen Ihrer Bedenkzeit), Zugvorschlag, Seitenwechsel, Hängespiel, Problemschach, Patreindruck und Speicherung, Blitzen usw. sprechen klar für CHESS.

DM 129,—

Hardware Uhr

Die Uhr wird ohne Lötarbeiten im Rechner eingebaut, so daß der ROM Port frei bleibt. Sie haben ständig die aktuelle Zeit und das aktuelle Datum! Die Batterie garantiert hohe Ganggenauigkeit und mindestens 3 Jahre Laufzeit bei voller Schaltjahrerkennung. Der sensationelle Preis für die anschlußfertige Uhr beträgt nur

DM 79,—

LOGIC ANALYSER

Endlich ein Logik Analyser, den sich jedermann leisten kann. Dabei können wir auf die Leistungsmerkmale von 5-10fach teureren Geräten hinweisen: 16 Kanäle bis 600 KHz, komfortable GEM Software, verschiedene Triggerungsmöglichkeiten usw. Das Gerät wird auf den Romport gesteckt und anschlußfertig mit Software geliefert.

DM 448,—

TV Modulatoren

Schließen Sie Ihren ST an den Farbfernseher an, und sparen Sie sich einen teuren RGB Monitor. Unsere Modulatoren bringen ein scharfes Bild mit allen Mischtönen und einen Tonausgang extern und einen über den Fernseher. Das heißt: Bestes Bild und bester Ton für alle Fernseher.

Typ A mit Monitorumschalter (ATARI-Fernseher)	DM 249,—
Typ B ohne Monitorumschalter	DM 239,—
Typ C FBAS Wandler für Geräte mit AV oder FBAS Eingang	DM 169,—

SYMPATIC PAINT

Der ATARI ST als Grafik Workstation der Superlative! Alle Funktionen, die für ein effektives und komfortables Zeichnen, Malen und Konstruieren erforderlich sind, wurden integriert. Der Benutzer hat nicht nur die Möglichkeit der Erstellung einer individuellen Grafiktoolbox, sondern auch eine Animationsselektion, mit der Bilder laufen lernen. Bilder können dabei mit dem G SCANNER und EASYTIZER eingelesen werden.

DM 298,—

Handbuch vorab: DM 30,—

G COPY

G COPY ist ein sehr vielseitiges Kopierprogramm, daß mit seinen zahlreichen Zusatzoptionen noch viel mehr kann: kopiert und formatiert alle Formate (912K-, 862K-, 416K-Disketten mit 10 oder 11 Sektoren und 80 - 83 Tracs) — optimale Kopiergeschwindigkeit bei 100% Datensicherheit — alle Programme ohne Hardwarekopierschutz werden kopiert

Neu: Bis Track 90

G Copy dient nur zur Erstellung von erlaubten Sicherheitskopien. **Raubkopien sind strafbar!**

Für alle ST mit Monochrom- und Farbmonitor DM 99,— —

Harddisk Help & extension

Wir wollen keinesfalls die Ängste um Ihre täglichen Daten schüren, aber sind Ihre Daten wirklich sicher? Wie schnell führt ein Headcrash zum Ende! Sie sollten auf ein bewährtes Backupsystem nicht länger verzichten. Auch sehr große Files können einzeln gesichert werden. Partition Backup, Tree Funktion, selected Backup und Diskoptimizer erhöhen die Effizienz dieses Programms. Auch bei der Geschwindigkeit wurden keine Kompromisse geschlossen.

DM 129,—

G DISKMON II

Zum Kontrollieren, Ausprobieren, Umkopieren, Manipulieren, Interessieren und Reparieren. Ein nützliches und wichtiges Tool, damit die Diskette kein Geheimnis bleibt. Unbeschränktes Lesen, Schreiben, Editieren, Kopieren, Suchen, Ersetzen, Formatieren usw.; Schnelldruck; Bootsektorscannung in Klartextdarstellung; interaktive Hex, Dez oder ASCII Bedienung.

DM 99,—

DISK HELP

Repariert Disketten und erlaubt in 99% der Fälle eine Wiederbelebung des gesamten Datenbestandes. Das Programm gehört neben jeden ST. Lassen Sie Ihre wichtigsten Daten nicht einfach von Laufwerken zerstören. DISK HELP ist einfach zu bedienen, korrigiert Lesefehler und rekonstruiert Files. Physikalische Fehler (Risse, Kaffee) können nicht behoben werden.

DM 79,—

M.A.R.S

Core Wars (Krieg der Kerne, Bericht in ST 11/87): Die neue Spielidee aus den USA. Für Strategen, die von Ballerspielen die Nase voll haben! Selbst programmierbare Viren kämpfen in Ihrem ST gegeneinander.

- grafische Kampfausgabe
- Trace Modus
- integrierter Editor
- voller Weltmeisterschaftsstandard
- ausführliche deutsche Dokumentation

DM 129,—

FAST SPEEDER

Für alle, die ihrem Massenspeicher keine Pause gönnen wollen und lange Wartezeiten satt haben. Bringen Sie Ihre Festplatte und Diskettenstation auf Trab! Im Extremfall erreichen Sie dabei Ramdiskgeschwindigkeit! Festplatten werden bis zu 200% beschleunigt, Diskettenlaufwerke bis zu 900%. FAST SPEEDER ist einfach zu handhaben. Der optimierte Spezialalgorithmus mit Cache Prinzip arbeitet für Sie unmerklich im Hintergrund.

DM 129,—

G DATA

Siemensstraße 16 4630 Bochum 1
Telefon: 0234 / 43 55 53
Verkaufsbüro:
Hattinger Str. 312, 4630 Bochum 1
Telefonische Bestellungen:
023 25 / 608 97

Schweiz: Computer Trend AG
Langstr. 31
CH-8021 Zürich

Österreich: Computershop Rittner
Hauptstr. 34
A-7000 Eisenstadt

Niederlande: Softpaquet International
Postbus 62 502702 AG
Zoetermeer

Alle Preise sind unverbindlich empfohlene Richtpreise

GEM Retrace Recorder

Der Knüller! Wie von Geisterhand bewegt sich die Maus, laufen Kommandos ab, werden Aktionen gestartet. Der GEM Retrace Recorder ermöglicht es, jegliche Aktion (Mausbewegung, Menüs, Klicks, Tastatur usw.) aufzuzeichnen, und auf Tastendruck jederzeit original zu wiederholen. Das heißt: GEM Makros erstellen für Aktionen, die ständig wiederholt werden (auch in jedem Programm); super Auto Boots erstellen; selbstlaufende Demos von jedem Programm ohne Programmeingriffe und Kenntnisse u.v.m. Die Anwendungsmöglichkeiten sind fast unbegrenzt!

DM 99,—

INTERPRINT II

Die universelle Anpassung für jeden Drucker hat einen integrierten Treiber, der auch alle Umlaute und Sonderzeichen nach Ihren Wünschen konvertiert. Dabei können mit Hilfe des Editors maßstabsgerechte Hardcopies auf allen Matrixdruckern und HP-Lasern erstellt werden. Auch ein Druckerspooler und unsere G RAMDISK II + sind integriert. Die Parameter sind speicherbar, und der komfortable Editor ermöglicht schnellste Anpassung.

DM 99,—

ohne Ramdisk DM 49,—

Roboterarm

Der semiprofessionelle 5achsige Roboterarm wird anschlußfertig für ST, AMIGA oder PC/AT geliefert. Für Schulungszwecke, Lehr- und Demonstrationszwecke oder einfach zum Spielen bestens geeignet. Technische Daten: Höhe ca. 54 cm, industriegelb, ca. 2,5 kg schwer, einfacher Anschluß an Drukkerport. Auch über Joysticks zu steuern.

DM 269,—

Anschlußfertig	DM 99,—
Spezialnetzteil	DM 10,—
Batteriesatz	

Hiermit bestelle ich:

- Informationsmaterial (DM 2,— beiliegend)
- 2 Demodisketten und Infomaterial (DM 10,— beiliegend)

COUPON

- per Nachnahme (ca. DM 6,50 Versandkosten)
- Verrechnungsscheck liegt bei (+ DM 5,— Versandkosten)

Absender:

ST-KONTOR

Der zweite Streich

Wie schon in der letzten Ausgabe erwähnt, werden vom Sybex-Verlag seit ca. einem halben Jahr verschiedene Module einer kaufmännischen Reihe vertrieben. Davon wurden in der letzten Ausgabe der ST-Kontor TOS-Manager und die ST-Kontor Kundenverwaltung vorgestellt. Um die ganze Sache abzurunden, beschäftigen wir uns heute mit den Anwendungen Lager & Fakturierung und der Finanzbuchhaltung.

Da uns allerdings von ST-Kontor Lager & Fakturierung noch keine endgültig lauffähige Version vorlag, beschränken wir uns auf die Beschreibung der zur Verfügung stehenden Möglichkeiten.

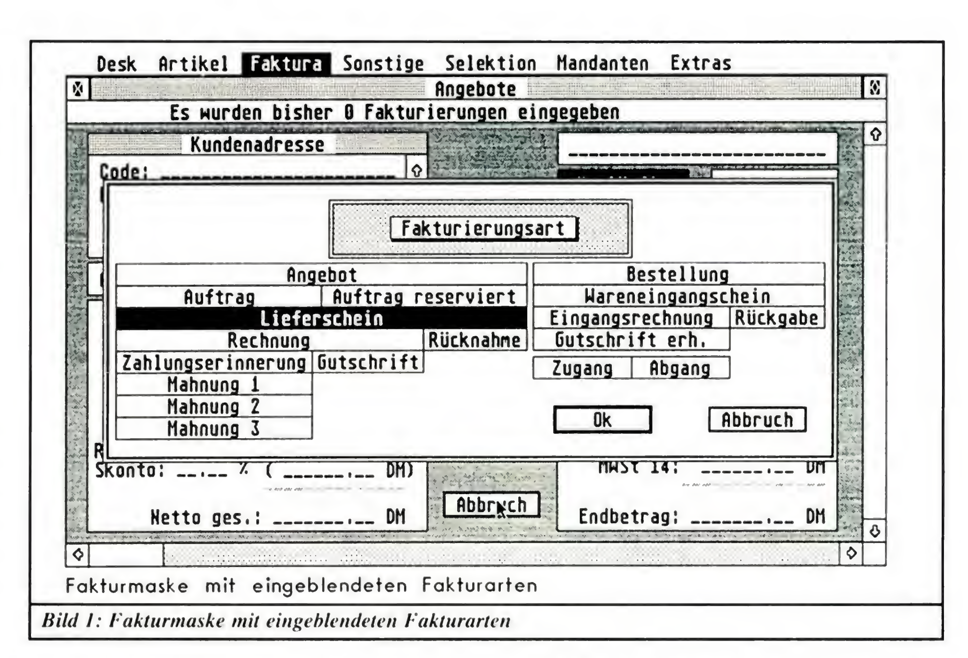


Bild 1: Fakturmaske mit eingeblendeten Fakturarten

ST-Kontor Lager & Fakturierung

Das Modul ST-Kontor Lager & Fakturierung ist ein Programm für Lagerverwaltung einschließlich Bestellwesen und ein Fakturprogramm. Falls man mit mehreren Programmen dieser Reihe arbeitet, wird auf einen gemeinsamen Datenbestand zugegriffen.

Hier nun ein kleiner Einblick in die Möglichkeiten, die das Modul bietet: Mit einer der wichtigsten Punkte einer Fakturierung ist die Erfassung der Artikel. Diese anlegen und bearbeiten ist die erste Funktion. Jeder Artikel gehört zu einer bestimmten Warengruppe, die separat angelegt wird. Für einen Artikel stehen 8 Datenmasken zur Verfügung, wovon zwei reine Informationswerte (Lagerbestände) enthalten, die nicht bearbeitet werden können. In den restlichen 6 Masken werden Textbausteine, Selektionskriterien, Lagerwerte und -bestände, Verkaufspreislisten, allgemeine Angaben, sowie Lieferanten, Bestellnummern, Lieferfristen und Einkaufspreise verwaltet.

Der nächste Punkt, Faktura 'Anlegen/ Bearbeiten/Buchhaltung', ermöglicht die eigentlichen Funktionen der Fakturierung. Die Belegarten reichen von Angebot über Auftrag (mit und ohne Lagerreservierung), Lieferschein, Rechnung, Warenrücknahme, Gutschrift, Zahlungserinnerung bis hin zur Mahnstufe. Zusätzlich können Bestellungen erstellt und auch diese weiterverarbeitet werden. Eine Belegart kann man vorläufig oder endgültig erstellen, woaus sich die Möglichkeit ihrer späteren Korrektur oder Umwandlung ergibt. Das Weiterverarbeiten von Belegarten ist möglich, d.h. aus einem Angebot wird ein Auftrag, aus einem Lieferschein eine Rechnung.

Für endgültig fakturierte Belege werden Buchungssätze erstellt, die in die FIBU eingelesen, überarbeitet und gebucht werden können.

Eine geschriebene Rechnung kann durch Betätigen der Tasten CONTROL P nach Bestimmen der Pfadangaben auf unterschiedliche Arten ausgegeben werden. Um ein Rechnungsformular zu ändern, ist zusätzlich ein Textprogramm notwendig. Empfehlenswert scheint für die Fakturierung hierbei 1st_Word plus zu sein, das allerdings nochmal mit 198,- DM den Geldbeutel belastet, wenn man ansonsten keine Texte mit dem Atari ST bearbeitet.

Unter dem Menüpunkt 'Sonstiges' haben sich noch ein paar kleine Besonderheiten versteckt. Die Funktion Inventur zeigt eine Übersicht über alle Artikel, für die eine Bestandsführung angelegt wurde. Der Gesamtwert des Lagers kann vom Programm aus den Artikeldaten vom Programm errechnet werden.

Nach Aufruf der Bildschirmkasse wird ein kleines Programmodul gestartet, mit der Lieferscheine und Rechnungen schnell fakturiert werden können. Die Eingabe erfolgt hauptsächlich über den Ziffernblock. Hier werden Artikel genauso einfach wie bei einer teuren Ladenkasse eingegeben und es wird eine Rechnung nach Bedarf geschrieben. Die Bild-

schirmkasse erstellt ein Kassenbuch für die täglich eingegebenen Verkäufe. Die Datensätze können später von der FIBU weiterverarbeitet werden.

Zurück zur normalen Fakturierung: Mit der Funktion Geschäftsjahreswechsel erfolgt eine Bereinigung der Dateien, d.h. die Jahresumsätze werden in die Maske 'Vorheriges Jahr' verschoben und die aktuellen Bestände in Warenanfangsbestände umgewandelt.

Mit der Barcode-Funktion besteht die Möglichkeit, ein weiteres Modul, den ST-Kontor Barcode-Drucker, aufzurufen, der für die Artikel Barcode-Etiketten (falls in der Artikelhauptmaske die Barcodenummer eingegeben wurde) drucken kann. Notwendig ist dafür ein Zusatzgerät, das in Kürze erscheinen soll.

Da diese Anwendung wie auch die ST-Kontor Finanzbuchhaltung mandantenfähig ist, müssen Mandanten angelegt und gewechselt werden können. Dies erfüllt der Menüpunkt Mandanten. Leider ist aber während der Verarbeitung nicht ersichtlich, für welchen Mandanten gerade fakturiert wird.

Das Modul ST-Kontor Lager & Fakturierung ist zu einem Preis von DM 398,- erhältlich. Die Programmdiskette und ein Handbuch mit ca. 200 Seiten gehören zum Lieferumfang. Die Beschreibung der einzelnen Funktionen ist teilweise nicht sehr anschaulich und gewöhnungsbedürftig, aber durch Ausprobieren kann diese Hürde genommen werden. Im Handbuch stehen neben der Programmbeschreibung und einigen Beispielen sämtliche Tastatur- und Mausbefehle, die man anfänglich immer wieder benötigt. Für die Erstellung einer eigenen Datenschnittstelle sind die Datenstrukturen im Handbuch mit abgedruckt.

ST-Kontor Finanzbuchhaltung

Die ST-Kontor Finanzbuchhaltung ist ein mandantenfähiges Finanzbuchhaltungsprogramm und entspricht dem neuen Bilanzrichtlinien-Gesetz. Das Erstellen einer Einnahme-Überschußrechnung ist mit diesem Programm nicht möglich.

Für einen einwandfreien Betrieb sind mindestens 1 MB Speicher, Monochrom-Monitor und ein doppelseitiges Laufwerk notwendig. Zum Arbeiten ist eine Festplatte mehr als nur sinnvoll, weil mit sehr vielen Dateien gearbeitet wird, und das Arbeiten mit Disketten schnell zu Bedienungsfehlern führen kann.

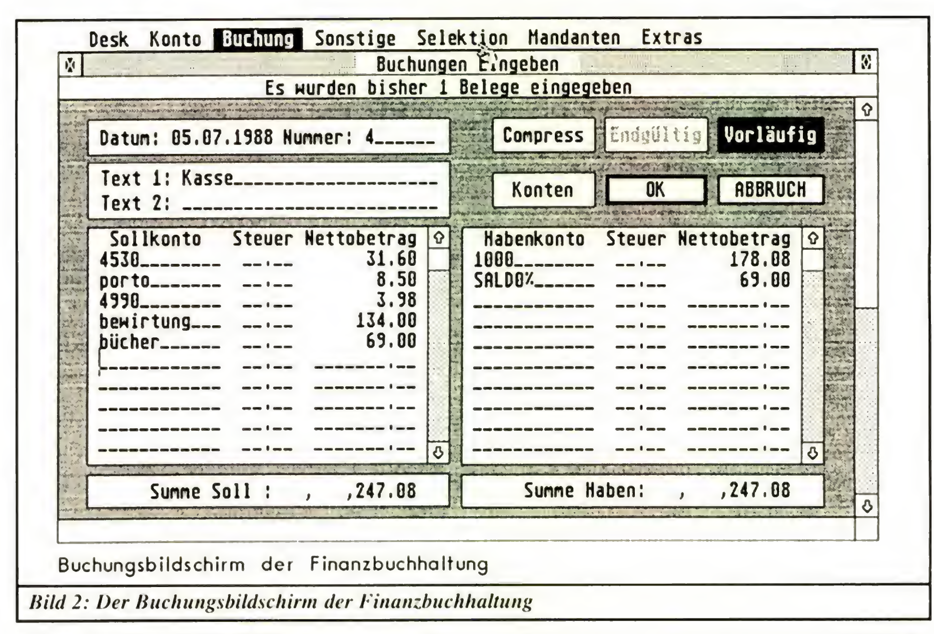


Bild 2: Der Buchungsbildschirm der Finanzbuchhaltung

Die Finanzbuchhaltung ist ein eigenständiges Programm. Für eine vernünftige Benutzung muß allerdings noch die ST-Kontor Kundenverwaltung vorhanden sein, damit auch der Zahlungsverkehr und das Mahnwesen über die FIBU abgewickelt werden können.

Der erste Menüpunkt befaßt sich mit dem Anlegen und Bearbeiten von Konten. Der Benutzer kann sich entscheiden, ob er sich seinen eigenen Kontenplan erstellt und danach die Konten eingibt, oder ob er den mitgelieferten Kontenplan, der den DATEV-Kontenrahmen SKR04 mit ca. 6000 angelegten Konten enthält, benutzt. Dieser Kontenplan befindet sich in komprimiertem Format auf der Diskette und muß vor der ersten Benutzung 'ausgepackt' werden.

Die Eingabe der Konten erfolgt in sechs Datenmasken. Ausgefüllt werden muß auf jeden Fall die erste, die zusätzlichen Masken nach Bedarf, wobei sich die letzte je nach Kontoart unterscheidet. Bei der Kontoart wird zwischen Sach-, Bank-, Personen-, Personal-, Waren-, Anlage-, Abschlußkonto und Kostenstelle unterschieden. Ein Konto kann durch eine Kurzbezeichnung aus maximal zwölf Zeichen angegeben werden, die später die Benutzung des Kontos ermöglicht.

Die Zusatzmasken gliedern sich auf in:

- eine Periodenübersicht für die monatlichen Soll und Haben Werte sowie die Salden für jeden Monat
- ein Kostenstellenverzeichnis mit bis zu zwölf Kostenstellen
- ein Notizfeld mit 12 Zeilen, die in Serienbriefe oder Listen eingelesen werden können
- die Selektionsmaske zur Selektion von Kontengruppen mit maximal 20 Selektionskriterien

Die Zusatzmasken der Kontoarten enthalten folgende Angaben:

Personenkonten: Adresse, Kontostand, Rabatt und Zahlungsbedingungen

Bankkonten: Bankverbindung und -adresse, 8 Pfadangaben

Anlagekonto: Angaben für Anlagespiegel, AfA-Art, Übersicht aktuelles Jahr und Vorjahr, Benutzer, Lieferant und Standort

Abschlußkonto: Abschlußarten (z.B.: Umsatzkostenverf. oder Gesamtkostenverf.), Wechselkonten

Kostenstellen: Angaben über Leiter evtl. Anschrift falls abweichend, Planzahlen können eingegeben werden, Abweichungen werden vom Programm errechnet

Der Druck der Konten wird über die Tastenkombination Control P erreicht. Hier kann zwischen verschiedenen Ausgabemedien (Monitor, Drucker, Datei) und Formaten (Liste, Brief, Etikett) gewählt werden.

Der nächste Menüeintrag ist der Buchung gewidmet. Hier lassen sich Buchungen eingeben und bearbeiten. Dabei besteht die Möglichkeit, Buchungssätze vom Programmpaket Lager & Fakturierung bzw. Lohn- und Gehalt zu übernehmen. Diese können bei Bedarf korrigiert und endgültig verbucht werden.

Der Buchungsbildschirm hat auf den ersten Blick einen ungewöhnlichen Aufbau, erweist sich aber nach näherer Betrachtung und ein paar Buchungen als recht praktisch. Die Bildschirmmaske ist in mehrere Blöcke unterteilt. Buchungsdatum und Belegnummer werden einge-

geben, dann folgt die Entscheidung, ob die Buchungen endgültig (können allerdings auch dann noch storniert werden) oder vorläufig erfaßt werden sollen. Die Eingabe erfolgt über den Ziffernblock, was das Ganze recht angenehm macht. Buchungsdatum und Belegnummer bleiben erhalten, man muß sie nicht immer von neuem eingeben. Ein schon eingegebener Buchungstext bleibt bestehen. Ich habe allerdings abspeicherbare Buchungstexte vermißt, die man bei Bedarf durch eine Ziffer oder ein Kürzel aufrufen kann.

Die beiden Buchungsblöcke splitten sich in Soll und Haben. Die Sollkonten stehen auf der linken Seite, was durch Kontonummer oder Kurzbezeichnung bewerkstelligt wird. Danach kommt die Angabe, ob aus diesem Betrag Mehrwertsteuer errechnet wird. Als letztes trägt man den Betrag (Nettobetrag) ein. Die eingegebenen Beträge werden anschließend zusammengefaßt auf die Habenseite unter Konto 'Saldo' übertragen. Das Habenkonto kann und muß nachträglich berichtigt werden. Es erscheinen bis zu 10 Buchungen auf dem Bildschirm, bei mehr als 10 hat man die Möglichkeit, zwischen den Buchungen zu scrollen.

Buchungen für ein Konto werden aufaddiert. Sie lassen sich, solange sie vorläufig sind, jederzeit unter 'Buchungen bearbeiten' korrigiert werden. Wenn alle Buchungen korrekt sind, kann man sie als endgültig erklären.

Auswertungen

Da die ST-Kontor Finanzbuchhaltung zur Erstellung der Abschlüsse mehrere Möglichkeiten bietet, muß der Anwender erst das Abschlußverfahren wählen. Unter anderem stehen ein Gesamtkosten- und ein Umsatzkostenverfahren zur Verfügung. Zusätzlich muß eine der 5 Abschlußarten: Bilanz, Gewinn- und Verlustrechnung, Anlagenspiegel, UST-Voranmeldung oder Kostenstellenübersicht ausgewählt werden. Zuletzt bestimmt man für die Abschlußart noch eine Periode, die einzelne oder beliebig viele Monate oder ein ganzes Jahr umfassen kann. Um in der ST-Kontor Finanzbuchhaltung den Zahlungsverkehr zu regeln, benötigt man unbedingt die ST-Kontor Kundenverwaltung. Zu Beginn der Bearbeitung wählt man ein Bankkonto aus. Am Zahlungsverkehr können nur Personenkonten teilnehmen.

Bei einem Geschäftsjahreswechsel werden alle Periodenumsätze gelöscht und die aktuellen Zahlen in das Feld 'Vorjahr' verlagert. Die Dateien Buchungen.Fib und Beleg.Fib werden in *BAK umbenannt und neu angelegt. Nun kann man das neue Jahr beginnen. Das alte Jahr läßt sich nun nicht mehr bearbeiten, es sei denn, man legt einen neuen Mandanten mit den alten Daten (im Handbuch beschrieben) an. Die Selektionskriterien sind mit denen der Fakturierung identisch. Auch in der Finanzbuchhaltung ist es möglich, Mandanten anzulegen und zwischen mehreren zu wechseln.

Die Extras bestehen aus den gleichen Funktionen wie in den anderen ST-Kontor Anwendungen:Es können Hilfstexte eingeblendet werden. Über Funktionstasten ist es noch möglich, auf Textverarbeitung umzuschalten, die Pfadangaben zu ändern, einen Notizblock einzublenden oder die Selektionskriterien auszudrucken. Auch eine Abfrage des vorhandenen Speicherplatzes und die Einstellung des Systems lassen sich von hier aus regeln.

Die ST-Kontor Finanzbuchhaltung kostet DM 498,-. Das Paket beinhaltet die Programmdiskette und ein gut 200 Seiten dickes Handbuch. Für ein solch umfangreiches Programm ist das Handbuch mit Beschreibung und Beispielen etwas dürftig ausgefallen. Es enthält auch die bei der Anwendung Lager & Fakturierung beschriebenen Befehle und Anhänge. Aufgrund des neuartigen Konzeptes dieser FIBU muß mit einer längeren, bzw. intensiveren Einarbeitungszeit gerechnet werden.

Von der GFE R. Becker KG wird für das gesamte ST Kontor-Paket die Möglichkeit von Einzelschulungen angeboten, die mit DM 190,- für eine Halbtagsschulung als angemessen preiswert anzusehen sind. Empfehlenswert sdind sie, wenn schnell mit dem Kontor-Paket gearbeitet werden soll, und eine längere Einarbeitungszeit im Endeffekt teurer wäre als die intensive Einführung in die Programme.

Fazit: Mit der ST-Kontor Reihe steht ein recht brauchbares Paket für den Atari ST im kaufmännischen Bereich zur Verfügung. Da die Version, die uns zum Testen vorlag, noch nicht der endgültig ausgelieferten Version entspricht, werden wahrscheinlich einige kleinere Fehler noch behoben werden, die bei der Benutzung aufgetreten sind.

Der TOS-Manager (s. ST 7/88) ist für die Anwendung der anderen Programme nicht notwendig, wahrscheinlich auch nicht besonders sinnvoll, da eigentlich alle wichtigen Funktionen genauso einfach vom Desktop aus erfolgen können.

Die Kundenverwaltung (s. ST 7/88) ist als Bestandteil für Fakturierung und FIBU notwendig: eine einfache Datenbank, die eine angenehme Erfassung der Kunden ermöglicht.

Die Fakturierung lag leider nicht in einer endgültigen Version vor. Eine Beurteilung, die über die Vorstellung der Funktionen hinausgeht, ist momentan also nicht möglich. Die Kasse macht einen robusten Eindruck. Es ist sicherlich ein gutes Programm für den Einsatz im Ladengeschäft.

Bei der FIBU sind die grundlegenden Ideen gut konzipiert, und auch die Möglichkeit einer preiswerten Schulung ist angenehm. Allerdings muß sich ein so komplexes Programm in der Praxis beweisen.

Karen Steger

```
Desk  Konto  Buchung  Sonstige  Selektion  Mandanten  Extras
                         Bearbeiten
  Erfasst am: 30.06.88     1000     Geändert am: 30.06.88

     Code: Kasse_______          Bankkonto____
   Nummer: 1000____
   Name 1: Kasse________________________________
   Name 2: _____________________________________
   Sammel: 9911240_  Steuer:   , 0 %   BWA: 0___

  Aktuelles Geschäftsjahr:       Vorhergehendes Geschäftsjahr:
   Summe Soll:    .   .100,00 DM   Summe Soll:    .   .   , 0 DM
  Summe Haben:    .   . 20,00 DM  Summe Haben:    .   .   , 0 DM
  Jahressaldo:    .   . 80,00 DM  Jahressaldo:    .   .   , 0 DM

   OK        ABBRUCH

Datenhauptmaske für Konten
```

Bild 3: Die Hauptdatenmaske der Konten

Einkaufsführer

Hier finden Sie Ihren Atari Fachhändler

1000 Berlin



DATAPLAY
Bundesallee 25 · 1000 Berlin 31
Telefon: 030/861 91 61

Computare
Keithstr. 18-20 • 1000 Berlin 30
030/21 390 21
186 346 com d





Vertragshändler
UNION ZEISS
Kurfürstendamm 57 • 1000 Berlin 15
Telefon 32 30 61

1000 Berlin



u. a. alphatronic, atari, commodore, dai, epson, sord mit pips, nec hard-/software nach maß — servicetechnik
Kurfürstendamm 121a, 1000 Berlin 31 (Halensee)
Telefon 030/891 1082

Computershop Behrendt
Reinecke, Tscheuschner GbR
Fürbringerstraße 26 · 1000 Berlin 61
Tel. (0 30) 6 91 76 66 · BTX (030) 6 91 76 66

COMPUTER-STUDIO
Schlichting
. . . die etwas andere Computerei
ATARI-Fachmarkt
MS-DOS Fachmarkt · NEC-Fachhandel
Katzbachstraße 8 · 1000 Berlin 61
☎ 030/7 86 43 40

2000 Hamburg

Bit Computer Shop
Osterstraße 173 · 2000 Hamburg 20
Telefon: 040/49 44 00

Createam
Computer Hard & Software
Bramfelder Chaussee 300 · 2000 Hamburg 71
Telefon Sa. Nr. 040/641 50 91

Hardware
Software
Beratung
Service



ATARI Systemfachhändler
Münsterstraße 9 · 2000 Hamburg 54
Telefon 040/56 60 1-1

ISYS
Software Shop
Bornstraße 1
2000 Hamburg 13

Gerhard u. Bernd Waller GbR
Computer und Zubehör-Shop
Kieler Straße 623
2000 Hamburg 54
Telefon (0 40) 570 60 07 + 570 52 75

RADIX Bürotechnik
Heinrich-Barth-Straße 13
2000 Hamburg 13
Telefon (0 40) 44 16 95
NEU: Software Shop



2000 Norderstedt



2120 Lüneburg

Sienknecht
Bürokommunikation
Beratung - Verkauf - Werkstatt

Heiligengeiststr. 20, 2120 Lüneburg
Tel. 0 41 31 / 4 61 22, Btx 40 24 22
Mo.-Fr. 9⁰⁰-18⁰⁰ und Sa. 9⁰⁰-13⁰⁰

2210 Itzehoe

Der Computerladen
Inhaber Ulrich Bübel Martin Kopplow

Coriansberg 2 · 2210 Itzehoe
Telefon (0 48 21) 33 90/91

2300 Kiel



Die Welt der Computer
Dreiecksplatz Nr. 7
2300 Kiel 1 · ☎ 04 31/56 70 42

2350 Neumünster



2390 Flensburg



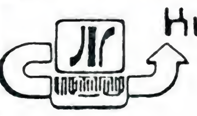
2800 Bremen



Faulenstraße 48—52
2800 Bremen 1
Telefon (04 21) 17 05 77

2850 Bremerhaven

HEIM- UND PERSONALCOMPUTER



Kurt Neumann
„Bürger" 160
2850 Bremerhaven
Tel. 04 71 / 4 20 06

HARDWARE · SOFTWARE · PAPIERWARE

2940 Wilhelmshaven

Radio Tiemann
ATARI-Systemfachhändler
Markstr. 52
2940 Wilhelmshaven
Telefon 0 44 21 - 2 61 45

2950 Leer



- HARDWARE-SOFTWARE
- EDV-SCHULUNG
- SYSTEM-ENTWICKLUNG
- EDV-BERATUNG
- ORGANISATION
- SERVICE-WARTUNG

Augustenstraße 3 · 2950 Leer
Telefon 04 91 - 45 89

3000 Hannover

COM DATA

Am Schiffgraben 19 · 3000 Hannover 1
Telefon 05 11 - 32 67 36

3000 Hannover



Großer Hillen 6 · 3000 Hannover 71
Telefon (0511) 52 25 79

DATALOGIC COMPUTERSYSTEME
ATARI ST-COMPUTER HARDWARE SOFTWARE
BERATUNG SERVICE VERKAUF
CALENBERGER STR. 26
3000 HANNOVER 1
TEL.: 0511 - 32 64 89

trend DATA Computer
IBM · EPSON · TRIUMPH ADLER
HEWLETT PACKARD · ATARI etc.

trendDATA Computer GmbH
Am Marstall 18-22 · 3000 Hannover 1
Telefon (05 11) 1 66 05-0

3040 Soltau

F & T Computervertrieb

Am Hornberg 1
(Industriegeb. Almhöhe)
3040 Soltau
Tel. 0 51 91 / 1 65 22

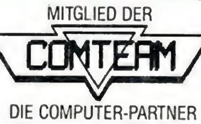
3150 Peine

Wieckenberg & Schrage GmbH
Computertechnik
Hard- u. Software
Woltorfer Str. 8, 3150 Peine
Tel. 0 51 71 / 60 52/3 o. 0 51 73 / 79 09

3170 Gifhorn

C OMPUTER
H AUS
G IFHORN

MITGLIED DER



DIE COMPUTER-PARTNER

INH. AXEL RITZ — POMMERNRING 38
D-3170 GIFHORN — TELEFON (0 53 71) 5 44 98
CELLER-BERLIN-BLUES MAILBOX — (0 51 41) 8 28 39

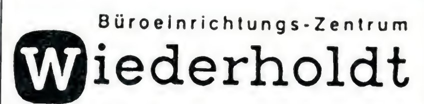
3300 Braunschweig

COMPUTER STUDIO
BRAUNSCHWEIG

Rebenring 49 - 50
3300 Braunschweig
Tel. (05 31) 33 32 77 / 78

3400 Göttingen

Büroeinrichtungs-Zentrum



3400 Göttingen-Weende
Wagenstieg 14 – Tel. 05 51 / 38 57-0

3470 Höxter



3500 Kassel

Hermann Fischer GmbH
autorisierter ATARI-Fachhändler
Rudolf-Schwander-Str. 5-13
3500 Kassel
Telefon (05 61) 70 00 00

4000 Düsseldorf

BERNSHAUS GmbH
Bürotechnik – Bürobedarf
Cäcilienstraße 2
4000 Düsseldorf 13 (Benrath)
Telefon 02 11 - 71 91 81

HOCO
EDV ANLAGEN GMBH
Flügelstr. 47
4000 Düsseldorf
Tel. 02 11 - 77 62 70

4050 Mönchengladbach



Hindenburgstr. 249
4050 Mönchengladbach
Tel. 0 21 61 - 1 87 64

4130 Moers



- Service-Center
- ATARI Fachhändler
- Hardware
- Software
- Erweiterungen

COP Computer Service GmbH
Essenberger Straße 2H · 4130 Moers
Telefon (0 28 41) 2 35 85

4150 Krefeld



- Festplatten
- Scanner
- Drucker
- BTX-Module
- Literatut
- Zubehör

COP Computer Service GmbH
Lewerentz-Straße 111 · 4150 Krefeld
Telefon (0 21 51) 77 30 42

4250 Bottrop

Megateam-Computer-Systeme

Kirchhellenerstraße 262
4250 Bottrop

4300 Essen

ATARI Systemfachhändler



KARSTADT Aktiengesellschaft
Limbecker Platz 4300 Essen 1
Tel.: (02 01) 17 63 99

4320 Hattingen

Ihre Tür zur Zukunft:



Hattingen, Große Weilstr. 18-20, Telefon (0 23 24) 20 94 77

4330 Mülheim



Computer und Bürotechnik
Vertriebsgesellschaft mbH
Dickswall 79 4330 Mülheim Telefon 0208/34034

Computer Hard- und Software auch im Leasing
Computerkurse für Anfänger und Fortgeschrittene

Tandon Computer · VICTOR



SEL-Fernkopierer

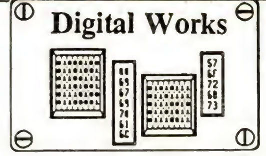
NEC · ATARI · OKI

4400 Münster



COMPUTER SYSTEME GMBH
Daimlerweg 39 - 4400 Münster
Telefon 02 51 / 71 99 75 - 9

4410 Warendorf



Computer-Fachhandel – Hardware & Software

Jörg Kellert – Helmut Müller GbR
Brünebrede 17 · 4410 Warendorf
Tel. 0 25 81 / 6 11 26

4422 Ahaus

ATARI · Epson · Fujitsu Molecular · NCR · Tandon · Schneider · Star

OCB

OCB-Computershop Wallstraße 3 4422 Ahaus Tel. 0 25 61/50 21	OCB-Hard- und Software Wessumerstraße 49 4422 Ahaus Tel. 0 25 61/50 21

4430 Steinfurt

ATARI SCHNEIDER STAR NEC SEIKOSHA PANASONIC EPSON

Computer
Büromaschinen
Service

Telefon 02551/2555

Tecklenburger Str. 27 · 4430 Steinfurt

4500 Osnabrück

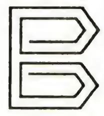
Heinicke-Electronic

Kommenderiestr. 120 · 4500 Osnabrück
Telefon 05 41 - 8 27 99

Wir liefern Micro-Computer seit 1978

4600 Dortmund

ATARI SYSTEM-Fachhändler



BÜRO STUDIO BOLZ

4600 Dortmund 1 · Brauhausstraße 4
Telefon (02 31) 5 27 7 13 - 16

Elektronik
Computer
Fachliteratur

ATARI-System-Fachhändler

4600 Dortmund 1, Güntherstraße 75, Tel. (02 31) 57 22 84

city-elektronik

ATARI Systemfachhändler



KARSTADT Aktiengesellschaft
Kampstraße 1 · 4600 Dortmund
Telefon (02 31) 5 43 91

4600 Dortmund



Atari, Genie, Schneider, Tandy, Brother, Star, Memorex, BASF, Verbatim
cc Computer Studio GmbH
Software-Hardware-Beratung
Service-Eilversand

Ihre Ansprechpartner:
v. Schablinski
Jan P. Schneider

Elisabethstraße 5
4600 Dortmund 1
T. 0231/528184 · Tx 822631 cccsd

4620 Castrop-Rauxel

R. Schuster Electronic
OBERE MÜNSTERSTR. 33-35 · (0 23 05) 3 77 0 · 4620 CASTROP-RAUXEL



System-Fachhändler

4650 Gelsenkirchen-Horst

MENTIS GmbH

Hard- und Software, Literatur
Bauteile, Service, Versand
Groß- und Einzelhandel

Poststraße 15 · 4650 Gelsenkirchen-Horst
Telefon (02 09) 5 25 72

4700 Hamm

computer center



4800 Bielefeld

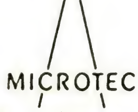
hardware
software
organisation
service

CSF

CSF COMPUTER & SOFTWARE GMBH
Heeper Straße 106–108
4800 Bielefeld 1
Tel. (05 21) 6 16 63

Carl-Severing-Str. 190
4800 Bielefeld 14

Telefon: 05 21 / 45 99-150
Telex : 9 37 340 krab d
Telefax: 05 21 / 45 99-123



MICROTEC
Software
Hardware
Beratung
Service

5000 Köln



AM RUDOLFPLATZ GmbH
5000 KÖLN 1
RICHARD-WAGNER-STR. 39
TEL. (02 21) 21 91 71

5010 Bergheim



EDV-Beratung · Organisation
Programmierung · Home/Personal-Computer
Software · Zubehör · Fachliteratur

Zeppelinstr. 7 · 5010 Bergheim
Telefon 0 22 71 - 6 20 96

5090 Leverkusen

Rolf Rocke
Computer-Fachgeschäft
Auestraße 1
5090 Leverkusen 3
Telefon 0 21 71 / 26 24

5220 Waldbröl



Waldstraße 1 · 5220 Waldbröl
☎ (0 22 91) 44 08/33 86

5300 Bonn

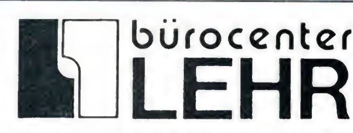


Gesellschaft für Computer- und Communicationstechnologie mbH
Hardware · Software · EDV-Zubehör
Telefon 02 28/22 24 08
COCO GmbH · Schumannstraße 2 · 5300 Bonn 1

5457 Straßenhaus

DR. AUMANN GMBH
Computer-Systeme
Schulstr. 12
5457 Straßenhaus
Telefon 0 26 34 - 40 81/2

5500 Trier



Güterstraße 82 · 5500 Trier
☎ 06 51 / 20 97 10
Fordern Sie unsere Zubehör-Liste an!

5540 Prüm

ATC COMPUTER
J. M. ZABELL
Ritzstraße 13 · Pf. 10 51
5540 PRÜM
– Tel.: 0 65 51 - 30 39 –

5600 Wuppertal

Jung am Wall
Wall 31—33
5600 Wuppertal 1
Telefon 02 02/45 03 30

COMPUTER FINKE



ATARI – SYSTEMFACHHÄNDLER
KIPDORF 22 · 5600 WUPPERTAL 1 · TEL. 0202 · 45 32 33
HARDWARE · SOFTWARE · ZUBEHÖR · SERVICE · SCHULUNGEN
ATARI
... wir machen Spitzentechnologie preiswert.

MEGABYTE
Computer Vertriebs GmbH
Friedrich-Engels-Allee 162
5600 Wuppertal 2 (Barmen)
Telefon (02 02) 8 19 17

5630 Remscheid

COM SOFT
Nordstraße 57 · 5630 Remscheid
Telefon (0 21 91) 2 10 33



Erfolgreich werben
Sprechen Sie mit uns.
Heim-Verlag ☎ (0 61 51) 5 60 57 BUF

5650 Solingen

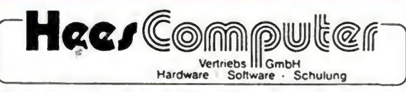
MegaTeam
Hardware — Software
Zubehör — Service
Rathausstraße 1-3 · 5650 Solingen 1
Telefon (02 12) 4 58 88

5800 Hagen



Vertragshändler Axel Böckem
Computer + Textsysteme
Eilper Str. 60 (Eilpezentrum) · 5800 Hagen
Telefon (0 23 31) 7 34 90

5900 Siegen



Siegen · Weidenauer Str. 72 · ☎ 02 71/7 34 95

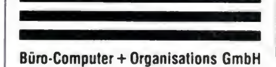
6000 Frankfurt

Müller & Nemecek
Kaiserstraße 44
6000 Frankfurt/M.
Tel. 0 69 - 23 25 44

WAIZENEGGER
Büroeinrichtungen
Kaiserstraße 41
6000 Frankfurt/Main
Tel. (0 69) 2 73 06 - 0



Eickmann Computer
Die Profis
Beratung, Service, Zubehör
In der Römerstadt 249
6000 Frankfurt/Main 90-Praunheim
Telefon (0 69) 76 34 09



Büro-Computer + Organisations GmbH



Oederweg 7-9
6000 Frankfurt/Main 1
☎ (0 69) 55 04 56 - 57
Commodore OKI ATARI TOSHIBA

6100 Darmstadt

Heim
Büro- und Computermarkt
Heidelberger Landstraße 194
6100 Darmstadt-Eberstadt
Telefon (0 61 51) 5 60 57

6200 Wiesbaden

COMPUTER TREFF
Computerbedarf und Software für ATARI, AMIGA, PC
Nettelbeckstraße 12
6200 Wiesbaden
Tel. (0 61 21) 40 43 02

6240 Königstein

KFC
COMPUTERSYSTEME
Wiesenstraße 18
6240 Königstein
Tel. 0 61 74 - 30 33
Mail-Box 0 61 74 - 53 55

6300 Gießen



6330 Wetzlar



Fachmarkt für Computer u. Unterhaltungs-electronic in Wetzlar,

Einkaufszentrum Bahnhofstraße, Tel. (0 64 41) 4 85 66

6400 Fulda

Schneider ATARI Commodore

WEINRICH

BÜRO · ORGANISATION

Ronsbachstraße 32 · 6400 Fulda

Telefon (06 61) 4 92 - 0

6457 Maintal

Landolt-Computer

Beratung · Service · Verkauf · Leasing

Wingertstr. 112
6457 Maintal/Dörnigheim
Telefon 0 61 81 - 4 52 93

6500 Mainz

:ELPHOTEC

Computer Systeme

Ihr Atari Systemhändler
mit eigenem Service-Center
Walpodenstraße 10
6500 Mainz
Telefon 0 61 31 - 23 19 47

6520 Worms

ORION

Computersysteme GmbH
Friedrichstraße 22
6 5 2 0 W O R M S
Tel. 0 62 41 / 67 57 - 67 58

6700 Ludwigshafen

MKV Computermarkt

Bismarck-Zentrum
6700 Ludwigshafen
Telefon 06 21 - 52 55 96

6720 Speyer

THEILLE Computersysteme

Gilgenstraße 4 · 6720 Speyer
Telefon (0 62 32) 772 16

6800 Mannheim



Computersysteme + Textsysteme

6800 Mannheim 24

Casterfeldstraße 74-76
☎ (06 21) 85 00 40 · Teletex 6 211 912

Computer-Center
am Hauptbahnhof GmbH

L 14, 16-17
6800 Mannheim 1
Tel. (06 21) 2 09 83 / 84

6900 Heidelberg

JACOM COMPUTERWELT

Hardware · Software
Schulung · Service

Mönchhofstraße 3 · 6900 Heidelberg
Telefon 0 62 21 / 41 05 14 - 550

HEIDELBERGER COMPUTER CENTER

Bahnhofstraße 1
6900 Heidelberg
Telefon 06221/27132

7022 L-Echterdingen

Autorisierter ATARI-System-Fachhändler

ATARI ST



Matrai Computer GmbH
Bernhauser Str. 8
7022 L. Echterdingen
☎ (07 11) 79 70 49

7030 Böblingen

Verkauf – Service – Software

Norbert Hlawinka
Sindelfinger Allee 1
7030 Böblingen
Tel. 0 70 31 / 22 60 15



COMPUTER CENTER

7100 Heilbronn

Unser Wissen ist **Ihr** Vorteil

Walliser & Co.

Mönchseestraße 99
7100 Heilbronn
Telefon (07 131) 6 00 48

7100 Heilbronn



Computer-Welt

Am Wollhaus 6
7100 Heilbronn
Tel. 0 71 31 - 6 84 01 - 02

7101 Flein

der **COMPUTERLADEN** von



In der Falterstraße
7101 Flein

Beratung, Service, Verkauf,
Software-Entwicklung
direkt beim Peripherie-Hersteller

7150 Backnang

Computer-Fans finden bei uns alles von:



7400 Tübingen

Werner Brock

COMPUTERSTUDIO

Poststraße 2—4 · D-7400 Tübingen
Tel. (0 70 71) 3 43 48 · Fax (0 70 71) 3 47 92

Autorisierter Systemfachhändler für:
ATARI, Schneider, Commodore, Panasonic, Kaypro, Sharp, NEC, OKI, STAR...

7410 Reutlingen

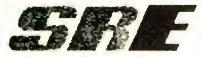
Werner Brock

COMPUTERSHOP

Federnseestr. 17 · 7410 Reutlingen
Tel. 0 71 21 - 3 42 87
Tx 1 72 414 024 RMI D · box:rmi:taisoft · Fax 0 71 21 - 33 97 79

Autorisierter Systemfachhändler für:
ATARI, Schneider, Commodore, Panasonic, Kaypro, Sharp, NEC, OKI, STAR,...

7450 Hechtingen



Gesellschaft für Datenverarbeitung mbH

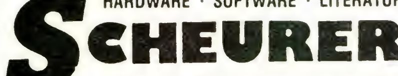
Computer · Drucker
Zubehör · Fachliteratur
Schloßplatz 3 · 7450 Hechingen
Telefon 0 74 71 / 1 45 07

7475 Meßstetten

Ihr ATARI-Systemhändler im Zollern-Alb-Kreis

HEIM + PC-COMPUTERMARKT

HARDWARE · SOFTWARE · LITERATUR



ATARI COMMODORE CUMANA DATA-BECKER
MULTITECH RITEMAN SCHNEIDER THOMSON

7475 Meßstetten 1 · Hauptstraße 10 · 0 74 31 / 6 12 80

7480 Sigmaringen

Wir stellen uns für Sie auf den Kopf!

Wir führen:
Hardware · Software · Zubehör
Zeitschriften · Bücher

Wir sind autorisierte Fachhändler von:
ATARI Schneider
Tandon

SOFT&EASY COMPUTER GMBH

7480 Sigmaringen
Am Rappgasse
Tel. 07571/12483

7968 Saulgau
Pfarrstraße 13
Tel. 07581/2598

7500 Karlsruhe

ERHARDT Am Ludwigsplatz
Am Ludwigsplatz · 7500 Karlsruhe 1 · Tel. (0721) 1608-0

MKV GMBH

Kriegsstraße 77
7500 Karlsruhe
Telefon (0721) 84613

7600 Offenburg

FRANK LEONHARDT ELECTRONIC
Ihr Fachgeschäft für Microcomputer · Hifi · Funk

In der Jeuch 3
7600 Offenburg
Telefon 0781/57974

7640 Kehl/Rhein



eigener Service · eigene Software

Badstrasse 12
Tel. 07807/822
Telex: 752913
7607 NEURIED 2

Filiale:
Hauptstrasse 44
Tel. 07851/1822
7640 KEHL/RHEIN

ELEKTRO-MÜNTZER GmbH

7700 Singen

U. MEIER
Computersysteme
7700 Singen-Htwl.
Am Posthalterswäldle 8
Telefon 07731-44211

7730 VS-Schwenningen

BUS BRAUCH & SAUTER
COMPUTER TECHNIK

Villinger Straße 85
7730 VS-Schwenningen
Telefon 07720/38071-72

7750 Konstanz

ATARI ★ PC's ★ SCHNEIDER

computer - fachgeschäft
rösler
Rheingutstr. 1 · ☏ 07531-21832

7800 Freiburg

CDS
EDV-Service GmbH

Windausstraße 2
7800 Freiburg
Tel. 0761-81047

PYRAMID COMPUTER GMBH
KARTÄUSERSTRASSE 59
D-7800 FREIBURG/BRST.
TELEFON 0761-382038

7890 Waldshut-Tiengen

hettler-data
service gmbh
Lenzburger Straße 4
7890 Waldshut-Tiengen
Telefon 07751/3094

7900 Ulm

HARD AND SOFT COMPUTER GMBH
Ulms großes Fachgeschäft
für BTX, Heim- u. Personalcomputer
Herrenkellergasse 16 · 7900 Ulm/Donau
Telefon 0731/62699

EDV-Systeme
Software-erstellung
Schulung

Systemhaus:
Frauenstraße 28
7900 Ulm/Donau
Tel. (0731) 28076
Telex 712973 csulm-d

COMPUTER STUDIO

7918 Illertissen

biotech gmbh
technische Informationsysteme
Computerladen

Marktplatz 13
7918 Illertissen
07303/5045

7980 Ravensburg

GRAHLE
Expert Grahle Computer
Eisenbahnstr. 33
7980 Ravensburg
Tel.: 0751/15955

Vertragshändler für ATARI, Schneider und Star

8000 München

star NEC



HDS-COMPUTER-VERTRIEBS GMBH
Am Klostergarten 1 · 8000 München 60
Tel. (089) 837063-64

ATARI Commodore

ludwig
COMPUTER + BÜROTECHNIK
COMPUTER · SOFTWARE · PERIPHERIE
BERATUNG · TECHN. KUNDENDIENST
INGOLSTÄDTER STRASSE 62L
EURO-INDUSTRIE-PARK · 8000 MÜNCHEN 45
TELEFON 089/3113066 · TELETEX 898341

schulz computer
Schillerstraße 22
8000 München 2
Telefon (089) 597339
Beratung · Verkauf · Kundendienst

8032 Gräfelfing



COMPUTER SYSTEME SCHULUNG
Am Haag 5
8032 Gräfelfing
Tel. 089-8545464,851043

8100 Garmisch-Partenk.

Uwe Langheinrich
Elektronik Center
Hindenburgerstr. 45
8100 Garmisch-Partenkirchen
Tel. 08821-71555
Bitte Gratisliste anfordern

8150 Holzkirchen

ATARI

Besuchen Sie uns!

Fordern Sie unseren Software-Katalog (520ST) an!



MÜNZENLOHER GMBH
Tölzer Straße 5
D-8150 Holzkirchen
Telefon: (0 80 24) 18 14

8170 Bad Tölz

Uwe Langheinrich Elektronik Center
Wachterstr. 3
8170 Bad Tölz
Tel. 0 80 41 - 4 15 65
Bitte Gratisliste anfordern

8330 Eggenfelden

Hot Space

Computer-Centrum
R. Lanfermann

Schellenbruckstraße 6
8330 Eggenfelden
Telefon 087 21 / 65 73

Altottinger Straße 2
8265 Neuotting
Telefon 086 71 / 7 16 10

8400 Regensburg

Zimmermann elektroland

8400 Regensburg
Dr.-Gessler-Str. 8
☎ 09 41 / 9 50 85

8390 Passau
Kohlbruck 2a
☎ 08 51 / 5 20 07

8423 Abensberg

COMPUTERVERSAND WITTICH
Tulpenstr. 16 · 8423 Abensberg
☎ 0 94 43 / 453



8500 Nürnberg

EINE IDEE ANDERS
KARSTADT NÜRNBERG AN DER LORENZKIRCHE



1. KLASSE EINKAUFEN IM WELTSTADTHAUS

hib
HIB Computer GmbH
Äußere Bayreuther Str. 57a – 59
8500 Nürnberg 10
Tel.: (09 11) 56 29 26 · Telex: 17 – 91 18 253
Teletex: 2627 – 91 18 253 · Telefax: (09 11) 51 30 40
Systemfachhändler für anspruchsvolle Computertechnik
Microsoft SONY ATARI brother TOSHIBA Zenith data systems EPSON

8520 Erlangen



Computerservice Decker
Meisenweg 29 - 8520 Erlangen
Telefon 0 91 31 / 4 20 76

Zimmermann elektroland

8520 Erlangen
Nürnberger Straße 88
Tel. (0 91 31) 3 45 68

8500 Nürnberg
Hauptmarkt 17
Tel. (09 11) 2 07 98

8600 Bamberg



Bamberg · Tel. 0951 / 2 78 08 - 09

8700 Würzburg

SCHOLL
BÜROTEAM

Hardware · Software
Service · Schulung

computer center
am Dominikanerplatz
Ruf (09 31) 3 08 08 - 0

8720 Schweinfurt

Uhlenhuth GmbH
Computer + Unterhaltungselektronik
Albrecht-Dürer-Platz 2
8720 Schweinfurt
Telefon 0 97 21 / 6 52 1 54

8900 Augsburg



Unser Plus: Beratung u. Service
Schwalbenstr. 1 · 8900 Augsburg-Pfersee
Telefon (08 21) 52 85 33 oder 52 80 87
Computer Vertriebs- und Software GmbH

Bei uns werben bringt

GEWINN



Sprechen Sie mit uns.
Heim Verlag 0 61 51 / 56057

BUF

8940 Memmingen

EDV-Organisation
Hard- + Software
Manfred Schweizer KG
Ulmer Str. 2, Tel. 0 83 31 / 1 22 20
8940 Memmingen

ÖSTERREICH

A-1020 Wien

W + H.
Computerhandel Ges m.b.H.

Förstergasse 6/3/2 · 1020 Wien
Tel. 02 22 - 35 09 68

A-1040 Wien

Ihr ST-Fachhändler in Wien
Computer-Studio
Wehsner Gesellschaft m.b.H.

1040 Wien - Paniglgasse 18-20
Telefon 02 22 - 65 78 08, 65 88 93

A-8010 Graz



SCHWEIZ

CH-1205 Geneve

PIMENT ROUGE
INFORMATIQUE S.A.
8, RUE DES MARAICHERS
1205 GENEVE TEL. 022/28 56 24

CH-1700 Fribourg

FRIDAT SA INFORMATIQUE
ehem. Softy Hard's Computershop

VOTRE SPECIALISTE
Rte des Grives 4
1700 Granges-Paccot/Fribourg
Tel. 0041 (0)37 26 66 28
Fax. 0041 (0)37 26 61 06

CH-2503 Biel



URWA ELECTRONIC
Computer Hard- und Software

Ihr ATARI ST Spezialist
in der Schweiz.
☎ 032/41 35 35

Bözingenstraße 133, 2504 Biel

CH-3084 Bern

Computer Corner
Ihr ATARI ST-Partner in Bern

Hard-Software Beratung
Midi-Schulung Service

Seftigenstr. 240 3084 Bern-Wabern
Zentrum Wabern Tel. 031/54 51 00

CH-4054 Basel

Wir sind Amiga- und ST-Freaks.

Unsere Öffnungszeiten:
Von Dienstag bis Freitag
9.30 – 12.30 und 14.00 – 18.30 und
am Samstag 9.30 – 16.00

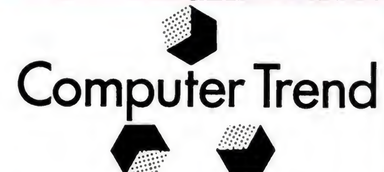
SYSAG
COMPUTER CENTER
Basel · Tel. 061 39 25 25 · Hulerstrasse 87 · 4054 Basel
Aarau · Tel. 064 22 63 33 · Kasernenstrasse 26 · 5000 Aarau

CH-4625 Oberbuchsiten

STECTRONIC M. Steck
Electronic-Computer-Shop

Hauptstr. 104/137
CH-4625 OBERBUCHSITEN
Tel. 062/63 17 27 + 63 10 27

CH-5430 Wettingen



Computer Trend
Zentralstrasse 93
CH-5430 Wettingen
Tel. 056 / 27 16 60
Telex 814 193 seco

CH-8001 Zürich

ADAG LASERLADEN
SEILERGRABEN 41 . 8001 ZÜRICH
MO-FR:12-18.30 TEL.01/251 49 34
DTP... LASERPRINTS ... ET CETERA!

CH-8006 Zürich

ADAG
COMPUTER-SHOP
UNIVERSITÄTSSTR. 25
8006 ZÜRICH TEL. 01/252 18 68

Computer-Center P. Fisch
Stampfenbachplatz 4
8006 ZÜRICH
01/363 67 67

CH-8021 Zürich



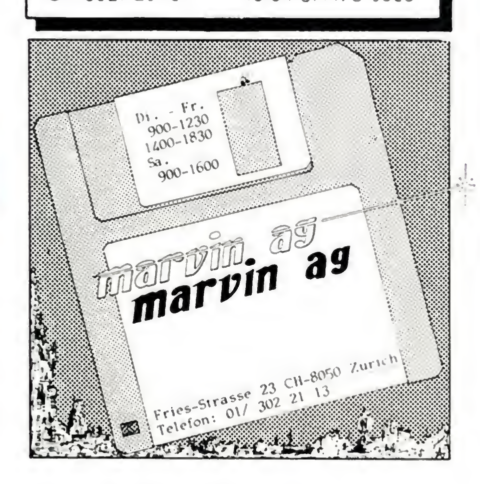
VILAN
Das Warenhaus der neuen Ideen

Ihr Computer-Fachhandel an der
Bahnhofstrasse 75 · Zürich

CH-8050 Zürich



Computer Trend
Langstrasse 31
Postfach
CH-8021 Zürich
Tel. 01 / 241 73 73
Telex 814 193 seco



CH-8052 Zürich



zentral in Seebach, Schaffhauser Str. 473, 8052 Zürich
Telefon: 01-302 26 00
Versandkatalog anfordern

CH-8200 Schaffhausen

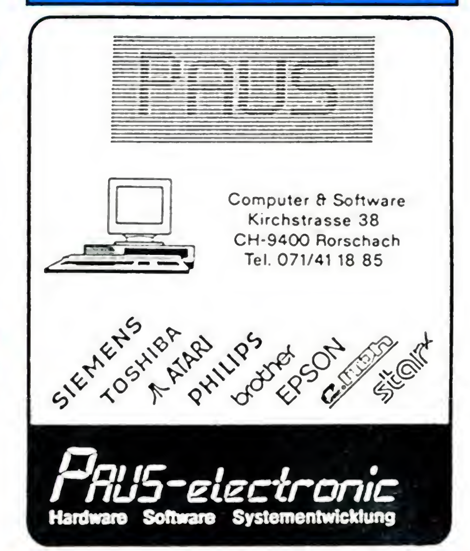
ZIMELEC
CAR-AUDIO
COMPUTER + ELECTRONIC
Bachstraße 28 · 8200 Schaffhausen
Tel.: 053/55224

Montag—Freitag 9.00—12.00
13.30—18.30
Samstag 9.00—16.00

CH-9000 St. Gallen

ADAG
COMPUTER-SHOP
TORSTR. 25
9001 ST.GALLEN
TEL. 071/25 43 42

CH-9400 Rorschach



LUXEMBURG



BILDUNG

Bisher wurde in allen möglichen Varianten besprochen, wie Bilder im ATARI ST-Speicher verwaltet werden. Inzwischen gibt es aber eine Unmenge von Grafik-Programmen wie DEGAS, IMAGIC, STAD, ART-DIRECTOR, DOODLE und viele mehr, so daß es, wie wir finden, an der Zeit ist, die Speicherformate dieser Programme zu beschreiben. Was nutzt das schönste Bild, wenn man es nicht vernünftig weiterverarbeiten kann, da das Dateiformat fehlt. Unter den gängigen Formaten befinden sich gepackte Formate, die wir auch nicht verschmähen wollen.

Aufgrund der Vielfalt der Dateiformate werden wir die Beschreibung auf mehrere ST-Ecken verteilen. In der heutigen Ausgabe wird noch einmal kurz auf die interne Grafikspeicherverwaltung des STs eingangen, um dann die einfachsten Dateiformate zu besprechen. Als Abrundung des Ganzen folgt dann ein Programm, mit dem man diese Dateien einlesen kann.

Der ATARI ST besitzt in seiner momentanen Version drei verschiedene Grafikauflösungen mit unterschiedlicher Anzahl von Farben. Da für jede Auflösung die gleiche Anzahl Bytes verwendet wird, muß mit steigender Auflösung zwangsläufig die Verfügbarkeit verschiedener Farben sinken. Bei der höchsten Auflösung (HIGH-Resolution) besitzt der ST 640 x 400 Punkte bei zwei Farben. Diese zwei Zustände der Farben (Bildpunkt AN/AUS) können in einem Bit abgespeichert werden, daher ergibt sich ein Speicherbedarf von 32000 Bytes (600 * 400 / 8 = 32000). Bei der mittleren Auflösung (MEDIUM-Resolution) werden 640 x 200 Punkte und damit vier Farben anhand zweier Bits kodiert, während die niedrige Auflösung 320 x 200 mit 16 Farben pro Punkt bietet, dafür aber vier Bits benötigt. Die hohe Auflösung ist in ihrer Speicherverwaltung relativ einfach zu verstehen, da ein Punkt in einem Bit kodiert ist und die Punkte hintereinander abgelegt sind. Das bedeutet, daß das Wort (zwei Bytes), das den ersten Punkt der zweiten Bildschirmzeile enthält, direkt hinter dem Wort steht, welches den letzten Punkt der ersten Zeile enthält. Die Berechnung der Adresse des Wortes, das unser Bildpunkt (Pixel) enthält, ist folgendermaßen durchzuführen:



Bild 1: Die hohe Auflösung

long Adresse;

Adresse = 2 (y * Worte pro Zeile + x/16) * Anzahl der Planes*

Der Rest der x/16-Division ergibt (indirekt) das Bit im Wort. Aber Achtung: Bit 15 ist Punkt 0 und Bit 0 ist Punkt 15, also ist die Bitnummer eines Pixels in einem Wort 15-x/16. Warum wir die Adresse auf Wortbasis errechnen, sehen wir gleich bei der MEDIUM- und LOW-Resolution.

Wir wissen, daß wir zur Darstellung von vier Farben zwei Bits benötigen, wobei es im ersten Moment naheliegt, diese für die Farbinformation benötigten Bits in einem Byte bzw. Wort hintereinander abzulegen. Dies wurde beim ST aber nicht getan. Hardwaremäßig ist es günstiger, die verschiedenen Bits eines Pixels in unterschiedlichen Bytes (oder Worten) an gleichen Bitpositionen unterzubringen. Prinzipiell kann man sich dies (ähnlich wie im Druckereigewerbe) als mehrere Folien vorstellen, die zusammen die Farbe ergeben (Bild 1 bis 3). Diese verschiedenen Ebenen nennt man Planes, so daß die High-Resolution ein Plane besitzt bzw. die Medium- zwei und die Low-Resolution vier davon.

In der Praxis gibt es nun zwei verschiedene Arten, die Planes im Speicher abzulegen. Die erste, für den Programmierer am einfachsten zu verstehende Methode ist, die einzelnen Planes wie völlig verschiedene Bildschirme zu behandeln, und daher einfach hintereinander abzulegen. Dieses Verfahren ist aber in der Hardwareumsetzung relativ umständlich. Einfach in Hardware umzusetzen, aber leider

schwieriger zu verstehen, ist das INTERLEAVED PLANE FORMAT, bei dem die Planes gemischt sind. Wie sie von oben noch wissen, wird die Bitinformation in einem Wort untergebracht. Hat man nun mehrere Planes, so folgt hinter Wort 0 der Plane 0 nicht das Wort 1 der Plane 0, sondern das Wort 0 der Plane 1 und dann das Wort 0 der Plane 2 usw., bis das jeweils erste Wort der vorhandenen Planes erfaßt ist. Der Nachteil ist, daß man ein wenig rechnen muß, wenn ein Punkt gesetzt werden soll. Am besten, Sie schauen sich noch einmal die Bilder 1 bis 3 und das Programmlisting 1 an, das an einem Beispiel zeigt, wie ein Punkt gesetzt wird. In ihm wird zunächst die Wortadresse berechnet und dann aufgrund des Teilerrestes x%16 (Modulo 16) das Bit an die richtige Position gerollt und ODER-verknüpft. Nur nebenbei: Die angegebene Methode des Bildschirmlöschens dient nur noch einmal zur Verdeutlichung, daß der Bildschirminhalt nichts anders als ein im Monitor sichtbar gemachter Speicher ist, den man auch löschen kann. Bitte löschen Sie Ihren Bildschirm nicht auf eine solche Methode - das ist sehr unsauber programmiert. Verwenden Sie zum Löschen des Bildschirms lieber eine VDI-Funktion.

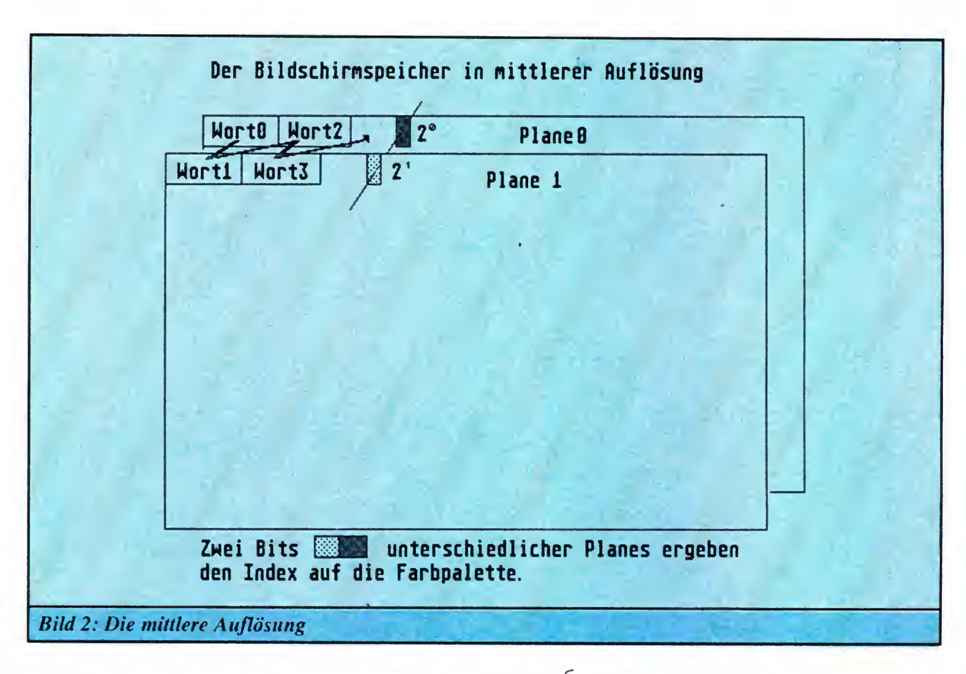


Bild 2: Die mittlere Auflösung

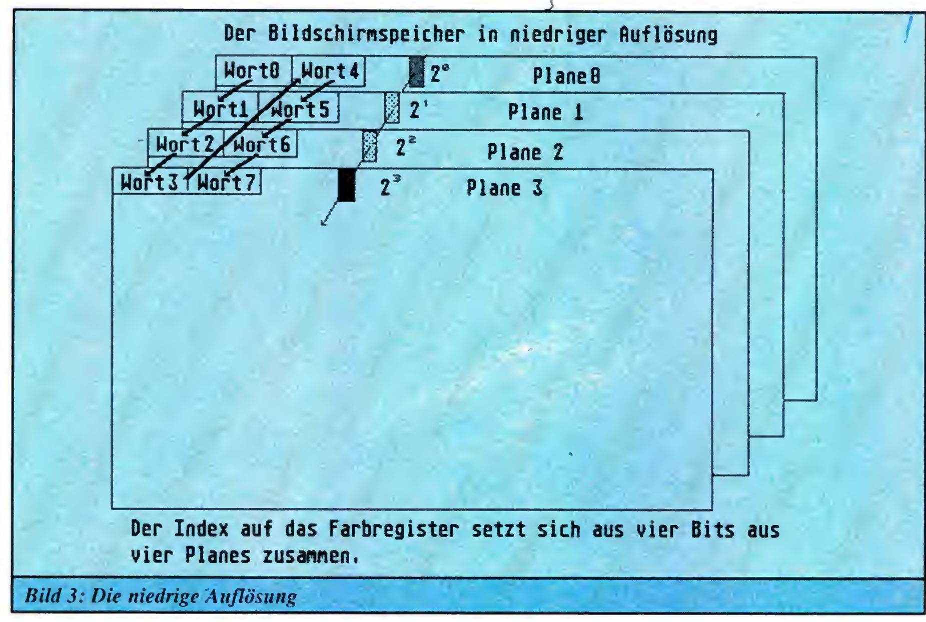


Bild 3: Die niedrige Auflösung

Farben wie ein Regenbogen...

hat der ST zwar nicht ganz, aber man hat doch immerhin 512 Farben zur Verfügung. Wie aber funktioniert das, wo wir doch nur höchstens vier Bits als Farbinformation zur Verfügung gestellt bekommen? Ganz einfach: Im ST befindet sich eine Liste, in denen die 16 aus den 512 ausgesuchten Farben stehen, die sie für ihre LOW-Resolution benutzen möchten. Findet der ST nun als Bitinformation im Grafikspeicher die Zahl 4, so schaut er in seiner Farbtabelle nach und nimmt diese dort abgespeicherte Farbe zur Bilddarstellung am Monitor. Eine solche Tabelle, die man in Fachkreisen auch LOOK-UP-Table nennt, hat gewisse Vorteile. Man kann, allein durch Austauschen der Werte in der Tabelle, die gesamten Farben des Bildes ändern. Wie wäre es zum Beispiel, wenn sie aus einem dunkelroten Abendhimmel ein wunderschönes, strahlendes Blau auf dem Bildschirm zaubern würden, ohne auch nur ein Bit im Grafikspeicher ändern zu müssen? Es ist sogar möglich, Farbverläufe, also Bildteile mit auf- oder absteigender Farbintensität durch Austauschen der Farbregisterinhalte zum Bewegen zu bringen. Kennen Sie die schöne ATARI-Demo, wo ein Vogel über eine Bucht fliegt und sich die Brandung bewegt? Die Animation der Brandung ist nur durch Verändern der Farb-Tabelle erreicht worden.

Das Geheimnis der 512 Farben

Die Farbtabelle liegt ab Adresse $FF8240 und kann und sollte über die XBIOS-Routine Setpalette(Integerfeld) gesetzt werden. Schauen wir uns zunächst aber an, wie eine Farbe definiert ist. Wenn ich Ihnen nun sage, daß die Farbinformation in einem Wort abgespeichert ist, so werden Sie sofort einwenden, daß damit ja 65536 Farben kodiert werden können. Richtig, es werden von den 16 Bits aber nur neun - entspricht 512 Farben - benutzt. Das Farbwort sieht folgendermaßen aus (x bedeutet nicht benutzt).

Farbwort: xxxx xRRR xGGG xBBB

Drei Nibbles (Halbbytes) werden für die Farbinformation benutzt, das vorderste nicht. Wie Sie vielleicht schon wissen, wird ein Farbbildpunkt auf einem Monitor (oder einem Fernseher) aus drei Punkten, nämlich einem blauen, einem grünen und einem roten, unterschiedlicher Intensität gewonnen. Sind alle drei Punkte aus, so gewinnt man Schwarz, werden alle Punkte mit höchster Intensität zum Leuchten gebracht, so erhält man die Farbe Weiß. Dieses RGB-Verfahren wird beim ATARI auch in dem Farbwort verwendet. Sie werden erkennen, daß in jedem Nibble die Intensität einer der drei Komplementärfarben eingestellt wird, wobei der ST in seiner bisherigen Ausführung nur Intensitäten zwischen 0 und 8 auswerten kann. Vielleicht wird es eine spätere Version des ST geben, die dann

DOODLE

Screenformat:	32000 Bytes Bilddaten, Bytes 1..32000, keine Farbpaletten
Auflösung:	LOW/MEDIUM/HIGH (*.DOO)
Dateilänge:	32000 Bytes

DEGAS normal

Screenformat:	32000 Bytes Bilddaten, Bytes 35 .. 32034, 32 Bytes Farbpalette, Bytes 3 .. 34, (32 Bytes Farbanimationsdaten, Bytes 32035 .. 32066)
Auflösung:	LOW (*.PI1), MED (*.PI2), HIGH (*.PI3)
Dateilänge:	32034 oder 32066 Bytes
Kennung:	1. Byte = 00 2. Byte = Auflösung (0=LOW, 1=MED, 2=HIGH)

NEO chrom

Screenformat:	32000 Bytes Bilddaten, Bytes 129 .. 32128, 32 Bytes Farbpalette, Bytes 5 .. 36
Auflösung:	nur LOW (*.NEO)
Dateilänge:	32128 Bytes

ART Director

Screenformat:	32000 Bytes Bilddaten, Bytes 1 .. 32000, 32 Bytes Farbpalette, Bytes 32001 .. 32032, weitere 15 Farbpaletten für Farbanimationen Bytes 32033 .. 32512
Auflösung:	nur LOW (*.ART)
Dateilänge:	32512 Bytes

Die Bildformate in der Übersicht

16 Intensitäten und damit 4096 Farben unterstützt. Ein kleiner Tip am Rande: Wenn Sie mal ein wenig mit den 512 Farben spielen möchten, so kann das jeder, der einen Farbmonitor (oder einen Fernseher) an seinen ST anschließt, denn mit dem Kontrollfeld ist es möglich, bequem die Farben des ST durch Slider (Schieber) einzustellen. Es sind drei Slider für rot, grün und blau vorhanden, wobei diese sich nur in acht Stufen bewegen lassen. (Warum bloß?)

Die Magie des Verschwindenlassens

Wollen wir die Grafiken abspeichern, so müssen zwei Informationen vorhanden sein, um das Bild vollständig rekonstruieren zu können: erstens der Bildschimspeicherinhalt mit den unterschiedlichen Planes und zweitens die Farbpalette. Ist man sich sicher, daß nur die Original-ATARI-Farbpalette benutzt worden ist, kann man auf das Abspeichern derselben verzichten.

Das hat Format

Verschiedene Grafikformate von verschiedenen Grafik-Programmen gibt es in Hülle und Fülle, so daß wir dieses Mal mit den einfachsten Formaten anfangen.

Das älteste und damit erste Malprogramm, das es für den ST gab, nennt sich DOODLE und ermöglichte ausschließlich das Zeichnen eines Bildes in der Original ATARI-Farbpalette. Daher werden als Datei auch nur einfach die 32000 Bytes eines Bildschirms abgespeichert und mehr nicht. Doodle-Bilder gibt es in allen Auflösungen, man kann sie aber nicht irgendwie unterscheiden - hier hilft nur ausprobieren. Der Extender des Dateinamens wird üblicherweise 'DOO', manchmal auch 'PIC', heißen.

Nach einiger Zeit folgte das Zeichenprogramm NEOCHROM, das zu damaliger Zeit durch die vielen Farben, die gleichzeitig im Auswahlfenster dargestellt wurden, auffiel. Eine Neochrom-Datei besitzt den Extender 'NEO' und ist 32128 Bytes lang. In den Bytes 5 bis 36 ist die Farbpalette abgespeichert, während sich die Bilddaten ab Byte 129 befinden und wie gewohnt 32000 Bytes benötigen. Neochrom arbeitet nur in der niedrigen Auflösung und speichert daher auch nur Bilder dieser Auflösung ab.

Das erste umfangreiche, verkaufte Malprogramm nennt sich DEGAS, benannt nach einem nicht unbekannten Maler. Mit DEGAS lassen sich Bilder in allen drei Auflösungen zeichnen und daher auch laden und speichern. Die Auflösung des abgespeicherten Bildes läßt sich am zweiten Byte der Bilddatei erkennen (das erste Byte ist Null) - dabei bedeutet 0 LOW-, 1 MEDIUM- und 2 HIGH-Resolution. Die folgenden 32 Bytes stellen wie gewohnt unsere Farbpalette dar und anschließend kommen die 32000 Bytes unseres Bildschirmspeichers. Da eine DEGAS-Datei aber anstatt der 32034 Bytes auch 32066 Bytes haben kann, fragt sich der schlaue Leser, was in den letzten 32 Bytes ist. In DEGAS ist es möglich, die oben beschriebene Farbanimation durch Vertauschen der Registerinhalte anzuwenden. Diese Farbinformation steht in den genannten 32 zusätzlichen Bytes. Die Datei hat abhängig von der Auflösung unterschiedliche Extender. Bei niedriger Auflösung verwendet DEGAS den Extender 'PI1', bei mittlerer 'PI2' und bei hoher 'PI3'.

Last not least schauen wir uns noch das Einfach-Format von ART-DIRECTOR an. Eine ART Director-Bilddatei besteht

aus 32512 Bytes, wobei die Bilddaten in den ersten 32000 Bytes enthalten sind. Die folgenden 32 Bytes enthalten wieder die bekannten Daten unserer Look-Up-Table. Der folgende Platz wird wie in DEGAS für Farbanimation benutzt, wobei im Gegensatz zu DEGAS bei ART-Director weitere 15 Farbpaletten abgespeichert werden! Wie ART-Director-Besitzern hinreichend bekannt ist, arbeitet dieses Programm nur in niedriger Auflösung und seine Dateien besitzen den Extender 'ART'.

Ein letztes Wort

Wie Sie sehen, sind die gängigsten Formate sehr einfach und damit auch ziemlich schnell in eigene Programme zu integrieren, zur einfacheren Handhabung haben wir die Daten noch einmal in einer Übersicht zusammengefaßt. Trotzdem haben wir noch ein Programm in PASCAL und C programmiert, damit die Vorgehensweise, ein Bild zu laden, klar wird. Bei der nächsten Wanderung durch die Epochen der Bildformate werden wir uns dann mit etwas aufwendigeren Methoden der Bildspeicherung beschäftigen, die es ermöglichen, die Datenmenge auf ein Minimum zu reduzieren (gepackte Bilddaten), oder ohne größere Probleme, Bilder unter Rechnern auszutauschen (Beispiel IFF-Format). Bis dahin wünsche ich noch frohes Laden, Speichern und natürlich Zeichnen von Bildern.

(SH / Jörg Drücker)

```
 1: /*****************************************************/
 2:  /*                                                   */
 3:  /* Zeichnen eines Punktes in beliebiger Auflösung    */
 4:  /*                                                   */
 5: /*****************************************************/
 6:
 7:  #include <osbind.h>
 8:  #include <gemdefs.h>
 9:
10:  #include <vt52.h>
11:
12:  int vdi_handle;
13:  int work_out[16], work_in[16];
14:
15: int intin[34],intout[16],ptsin[16], ptsout[16],
        contrl[10];
16:
17: /*****************************************/
18:  /* Diese Programm wird durch Anfahren der */
19:  /* linken oberen Ecke beendet             */
20: /*****************************************/
21:
22:  main()
23:  {
24:     int loop, x,y, wort;
25:     unsigned int *wortadr;
26:     int *basis;
27:     long *speicher;
28:     int status;
29:     int planes;
30:
31:
32:     appl_init();             /* Applikation anmelden */
33:
34:     open_vwork();            /* VDI initialisieren   */
35:
36:     planes = 1<<(2-Getrez()); /* Anzahl der Bits pro
                                    Pixel */
37:
38:     basis = (int *) Logbase();
39:     speicher= (long*) basis;
40:
41:
42:     graf_mouse(M_OFF,0L);
43:     loop=8000;    /* 8000 Langwörter = 32000 Bytes */
44:     while(loop)   /* Bildschirmspeicher löschen */
45:      {
46:        *speicher++ =0L;   /* Langwort löschen */
47:        loop--;
48:      }
49:     graf_mouse(M_ON,0L);
50:
51:
52:      do
53:      {
54:       vq_mouse(vdi_handle,&status,&x,&y);
                       /* Mauskoordinate holen */
55:        if (status)  /* Maustaste gedrückt  */
56:         {
57:          graf_mouse(M_OFF,0L); /* Maus ausschalten */
58:
59:          wort=(y*40+x/16)*planes;
60:
61:          wortadr=(unsigned int *) (basis+wort);
62:          *wortadr|= ((unsigned int )0x8000)>>(x%16);
63:
64:          graf_mouse(M_ON,0L); /* Maus anschalten */
65:         }
66:      }while(x || y);          /* bis Koordinate 0,0 */
67:
68:     v_clsvwk(vdi_handle);  /* VDI abmelden */
69:     appl_exit();           /* Applikation abmelden */
70:  }
71:
72: open_vwork() /* Initialisieren der VDI-Workstation */
73:  {
74:     int i;
75:
76:     for (i=0; i<10; work_in[i++]=1);
77:     work_in[10]=2;
78:    v_opnvwk(work_in, &vdi_handle, work_out);
79:  }
```

Listing 1: So setzt man einen Punkt in beliebiger Auflösung

```
 1: program BILDEINLESEN ( input, output );
 2:
 3: { *-------------------*
 4:
 5:   Demonstrationsprogramm zum Einlesen
 6:   von verschiedenen Bildformaten.
 7:   ------------------
 8:
 9:    Version 1.0
10:
11:    verfasst am 7-6-1988 von Jörg Drücker
12:
13:    Copyright (c) 1988 by IMAGIC GRAFIK.
14:
15:    *-------------------* }
16:
17:  {$P-, keine Pointerchecks ! }
18:
19:  { globale Konstanten:
20:    ---------- }
21:
22:  const  PICTURELEN = 32000; { ST Bildschirmgrösse }
23:
24:  { globale Datentypen:
25:    ---------- }
26:
27: type DATA_POINTER = ^byte; {universeller Pointertyp}
28:
29:      COLOUR_DATA = packed array [0 .. 15] of integer;
30:      COLOUR_PTR  = ^COLOUR_DATA;
31:
32:      BYTE_ARRAY = packed array [ 1..maxint ] of byte;
33:
34:        { Maximal 32767 Bytes, für Bilddaten voll
             ausreichend }
35:
36: { bekannte Bildformate:
37:   ---------- }
38:
39:   PICTURE_TYPES = (  P_DEGAS, { Degas unkomprimiert }
40:                      P_DOODLE, { Doodle/Screenformat }
41:                      P_NEO,     { Neochrome Format }
42:                      P_ART,     { ART Director }
43:                      P_UNDEF ); { unbekanntes Format }
```

```
 44:
 45: { globale Variablen:
 46:   ————————— }
 47:
 48: var   LESEBUFFER  : record case boolean of
 49:
 50:                       false : ( DATA :DATA_POINTER );
 51:                       true  : ( BLOC :^BYTE_ARRAY )
 52:
 53:                     end;
 54:
 55:
 56:       BILDSCHIRM  : DATA_POINTER;
 57:
 58:       SYSTEMFARBE,
 59:       BILDFARBE   : COLOUR_PTR;
 60:
 61:       EXTERNDATEI : packed file of BYTE_ARRAY;
 62:       DATEINAME   : string [ 255 ];
 63:
 64:       BILDTYP     : char;
 65:       PIC_TYPE    : PICTURE_TYPES;
 66:
 67: { Systemfunktionen:
 68:   ———————— }
 69:
 70: function physbase : DATA_POINTER;
 71:
 72: { Physikalische Bildschirmaddresse ermitteln }
 73:
 74:          xbios ( 2 );
 75:
 76: procedure setpalette ( palette : COLOUR_PTR );
 77:
 78: { Farbpalette setzen }
 79:
 80:          xbios ( 6 );
 81:
 82: { Hilfsprozeduren in PASCAL:
 83:   ——————————— }
 84:
 85: function ADDR_OFFSET ( D : DATA_POINTER;
 86:                        OFFSET : long_integer )
                                  : DATA_POINTER;
 87:
 88: { berechnet Offset zur gegebenen Addresse D hinzu. }
 89:
 90: var X : record case boolean of
 91:
 92:              false : ( DATA : DATA_POINTER );
 93:              true  : ( ADDR : long_integer )
 94:
 95:         end;
 96:
 97: begin
 98:
 99:       X.DATA := D;                  { Pointer kopieren  }
100:       X.ADDR := X.ADDR + OFFSET;    { Offset addieren   }
101:       ADDR_OFFSET := X.DATA         { Ergebnis zuweisen }
102:
103: end; { ADDR_OFFSET }
104:
105: procedure TRANSFER (SOURCE, DESTINATION :DATA_POINTER;
106:                     AMOUNT : integer );
107:
108: { Byteweises kopieren beliebiger Speicherbereiche }
109:
110: var S, D : record case boolean of
111:
112:              false : ( DATA : DATA_POINTER );
113:              true  : ( BLOC : ^BYTE_ARRAY )
114:
115:            end;
116:
117:      I : integer;
118:
119: begin
120:
121:       S.DATA := SOURCE;          { Pointer zuordnen }
122:       D.DATA := DESTINATION;
123:
124:       for I := 1 to AMOUNT do {Anzahl Bytes kopieren}
125:
126:           D.BLOC^ [ I ] := S.BLOC^ [ I ]
127:
128: end; { TRANSFER }
129:
130: procedure SYSPALET ( C : COLOUR_PTR );
131:
132: { Hole aktuelle Farb-Palette }
133:
134: var I : integer;
135:
136: function setcolor (register, value:integer):integer;
137:          xbios ( 7 );
138:
139: begin
140:
141:       for I := 0 to 15 do
142:
143:           C^ [ I ] := setcolor ( I, -1 )
                          { Farbe nur holen }
144:
145: end; { SYSPALET }
146:
147: procedure HOLE_BILDDATEN (READBUF,DEST: DATA_POINTER;
148:                           PIC_TYPE:PICTURE_TYPES);
149:
150: { Hole Bildinhalt, je nach Bildformat }
151:
152: begin
153:
154:       case PIC_TYPE of
155:
156:         P_DEGAS  : TRANSFER (ADDR_OFFSET(READBUF,34),
157:                              DEST,
158:                              PICTURELEN);
159:
160:         P_DOODLE : TRANSFER (READBUF,
161:                              DEST,
162:                              PICTURELEN);
163:
164:         P_NEO    : TRANSFER(ADDR_OFFSET (READBUF,128),
165:                              DEST,
166:                              PICTURELEN);
167:
168:         P_ART    : TRANSFER (READBUF,
169:                              DEST,
170:                              PICTURELEN)
171:
172:       end { case }
173:
174: end; { HOLE_BILDDATEN }
175:
176: procedure HOLE_FARBEN ( READBUF : DATA_POINTER;
177:                         FARBEN  : COLOUR_PTR;
178:                         PIC_TYPE : PICTURE_TYPES );
179:
180: { Hole Farbpalette, je nach Bildformat }
181:
182: var COL : record case boolean of
183:
184:              false : ( DATA : DATA_POINTER );
185:              true  : ( CPTR : COLOUR_PTR )
186:
187:           end;
188:
189: begin
190:
191:       COL.CPTR := FARBEN;  { Pointer gleichsetzen }
192:
193:       case PIC_TYPE of
194:
195:         P_DEGAS : TRANSFER (ADDR_OFFSET(READBUF,2),
196:                             COL.DATA,
197:                             32);
198:
199:         P_DOODLE : SYSPALET (COL.CPTR);
                        { aktuelle Palette holen }
200:
201:         P_NEO    : TRANSFER (ADDR_OFFSET(READBUF,4),
202:                             COL.DATA,
203:                             32);
204:
205:         P_ART    : TRANSFER (ADDR_OFFSET(READBUF,
                                 PICTURELEN),
206:                             COL.DATA,
207:                             32)
208:
209:       end { case }
210:
211: end; { HOLE_FARBEN }
```

```
212:
213: { Hauptprogramm:
214:   ——————— }
215:
216: begin
217:
218:      io_check ( false );  { eigene Fehlerbehandlung }
219:
220:      new ( LESEBUFFER.BLOC );  { Arbeits - Bereiche
                                     anlegen }
221:      new ( SYSTEMFARBE );
222:      new ( BILDFARBE );
223:
224:      BILDSCHIRM := physbase;   { Addresse des phys.
                                     Bildschirms }
225:      SYSPALET ( SYSTEMFARBE ); { Ursprüngliche
                                     Farbpalette sichern }
226:
227:      loop
228:
229:           writeln ( chr ( 27 ), 'E' );  { Bildschirm
                                             löschen }
230:
231:           write   ( 'Bitte Bildnamen eingeben: ' );
232:           readln  ( DATEINAME );
233:
234:      exit if length ( DATEINAME ) = 0;
235:
236:           reset ( EXTERNDATEI, DATEINAME );
237:
238:           if ( io_result = -33 ) or ( io_result = 9 )
               then begin
239:
240:              writeln ( chr ( 7 ) );  { BEEP }
241:              writeln ( 'Datei ', DATEINAME,
                            ' existiert nicht !' );
242:              writeln;
243:              write   ( 'drücke <RETURN> um
                            fortzufahren ...' );
244:              readln
245:
246:           end
247:           else begin
248:
249:              repeat
250:
251:                 writeln;
252:                 writeln ( 'Bildtyp eingeben:' );
253:                 writeln ( '      A = DEGAS Normal' );
254:                 writeln ( '      B = DOODLE' );
255:                 writeln ( '      C = NEO Chrome' );
256:                 writeln ( '      D = ART Director' );
257:
258:                 write   ( '> ' );
259:                 readln  ( BILDTYP );
260:
261:                 case BILDTYP of
262:
263:                      'A','a' : PIC_TYPE := P_DEGAS;
264:                      'B','b' : PIC_TYPE := P_DOODLE;
265:                      'C','c' : PIC_TYPE := P_NEO;
266:                      'D','d' : PIC_TYPE := P_ART;
267:
268:                     otherwise : PIC_TYPE := P_UNDEF
269:
270:                 end
271:
272:             until PIC_TYPE <> P_UNDEF;
273:
274:             LESEBUFFER.BLOC^ := EXTERNDATEI^;
                { Daten zuordnen }
275:
276:             { Cursor ausschalten:
277:               ————————— }
278:
279:             write ( chr ( 27 ), 'f' );
280:
281:             { Bildinhalt beispielsweise zeigen:
282:               ———————————————— }
283:
284:             HOLE_FARBEN ( LESEBUFFER.DATA, BILDFARBE,
                              PIC_TYPE );
285:             setpalette  ( BILDFARBE );
286:             HOLE_BILDDATEN ( LESEBUFFER.DATA,
                                 BILDSCHIRM, PIC_TYPE );
287:
288:                readln; { warte auf Tastendruck }
289:
290:                setpalette ( SYSTEMFARBE );
291:                { Ursprüngl.Palette wiederherstellen }
292:
293:                { Cursor wieder einschalten:
294:                  ————————————— }
295:
296:                write ( chr ( 27 ), 'e' )
297:
298:             end
299:
300:        end; { loop }
301:
302:        writeln;
303:        writeln ( 'Adieu' )
304:
305: end.
```

Listing 2: Das Laden verschiedener Bildformate in PASCAL

```
 1: /*****************************************************/
 2: /*                                                   */
 3: /*   Programm zum Einlesen diverser Bildformate      */
 4: /*   wie DEGAS,DOODLE,ART Director und NeoChrome     */
 5: /*                                                   */
 6: /*****************************************************/
 7:
 8: #include <osbind.h>
 9:
10: #define PICTURELEN 32000L
11: #define P_DEGAS 1
12: #define P_DOODLE 2
13: #define P_NEO 3
14: #define P_ART 4
15:
16: long bildschirm;  /* Enthält Adresse des Bildschirms */
17: char *buffer;  /* Adresse, an d.Daten geladen werden */
18: int systemfarbe[16], bildfarbe[16];
19:
20: char pfad[64], fname[64];
21:
22: main()
23: {
24:   int taste, i;
25:   int fd, pic_type;
26:
27:   appl_init();
28:
29:   buffer= (char*) Malloc(32550L);/* Speicher reserv. */
30:   if (buffer==(char*)-1)          /* Kein Speicher da */
31:   {
32:     appl_exit();
33:     return;                       /* Programm beenden */
34:   }
35:
36:   bildschirm = Physbase();  /* Adresse d.Bildschirms */
37:   syspalet(systemfarbe);    /* Original-Palette holen */
38:
39:   printf("\033E\n");        /* Bildschirm löschen */
40:
41:   strcpy(pfad,"\*.*");      /* Anfangspfad */
42:   fname[0]=0;
43:
44:   fsel_input(pfad,fname,&taste); /* Fileselektor */
45:
46:
47:   if (taste)    /* OK gedrückt */
48:   {
49:     i=0;
50:     while(pfad[i]) i++; /* Bis ans Ende */
51:         /* und dann Wildcards wegwerfen */
52:     while(i && pfad[i]!=':' && pfad[i]!='\\') i–;
53:     strcpy(&pfad[i+1],fname); /* u.Filename anhängen */
54:
55:
56:     fd = Fopen(pfad,0);        /* Datei öffnen */
57:
58:     if (fd>0)                  /* kein Fehler */
59:     {
60:     pic_type=form_alert(0,"[1][Bitte Bildtyp eingeben]
                            [DEGAS|DOODLE|weiter]");
61:
62:       if (pic_type==3) /* 'Weiter' angewählt */
```

```
 63:          pic_type=2+form_alert(0,"[1][Bitte Bildtyp
                       eingeben][NEOchrome|ART-Director]");
 64:
 65:                        /* Mal 32550 Bytes laden */
 66:        Fread(fd,32550L,buffer);
 67:                        /* Palette aus Daten ziehen */
 68:       hole_farben(buffer,bildfarbe,pic_type);
 69:
 70:        Setpallete(bildfarbe); /* und setzen */
 71:       hole_bilddaten(buffer,bildschirm,pic_type);
 72:         gemdos(7);          /* Tastendruck */
 73:       Setpallete(systemfarbe); /* Org.-Palette setzen */
 74:
 75:         Mfree(buffer);          /* Speicher freigeben */
 76:         Fclose(fd);             /* Datei schliePen    */
 77:       }
 78:      else
 79:       form_alert(1,"[1][Beim Lesen ist|ein Fehler
                    aufgetreten][ Abbruch ]");
 80:    }
 81:    appl_exit();                /* TschüP */
 82:   printf("Adieu\n");
 83:
 84:  }
 85:
 86: /*******************************************************/
 87:  /* Diese Routine kopiert einen Speicherbereich       */
 88:  /* von Source nach Destination mit amount Bytes      */
 89: /*******************************************************/
 90:
 91:
 92: transfer(source, destination, amount)
 93:  char *source, *destination;
 94:  long amount;
 95:  {
 96:    while(amount—)
 97:      *destination++ = *source++;
 98:  }
 99:
100:
101: /*******************************************************/
102:  /* Die Routine ermittelt die momentane Palette       */
103:  /* und schreibt sie in c                             */
104: /*******************************************************/
105:
106:
107: syspalet(c)
108: int *c;  /* zeigt auf Integerfeld von 16 Eintraegen */
109:  {
110:    int i;
111:
112:    for (i=0; i<16; i++)
113:      *c++ = Setcolor(i,-1);    /* Farbe nur holen */
114:  }
115:
116:
117: /*******************************************************/
118:  /* Routine holt aus den Daten die Bildaten je nach    */
119:  /* Bildformat heraus                                  */
120: /*******************************************************/
121:
122:
123: hole_bilddaten( readbuf, dest, pic_type)
124:  char *readbuf, *dest;
125:  int pic_type;
126:  {
127:    switch(pic_type)
128:     {
129:       case P_DEGAS:
130:     transfer(readbuf+34,dest,PICTURELEN);
131:       break;
132:
133:       case P_ART:
134:       case P_DOODLE:
135:     transfer(readbuf,dest,PICTURELEN);
136:       break;
137:
138:       case P_NEO:
139:     transfer(readbuf+128,dest,PICTURELEN);
140:       break;
141:
142:     }
143:  }
144:
145: *******************************************************/
146:  /* Hier wird die Palette aus den Daten aussortiert    */
147: /*******************************************************/
148:
149:
150: hole_farben(readbuf, farben, pic_type)
151:  char *readbuf;
152:  int *farben;
153:  int pic_type;
154:  {
155:    switch(pic_type)
156:     {
157:       case P_DEGAS:
158:     transfer(readbuf+2,farben,32L);
159:       break;
```

Listing 3: Das Laden verschiedener Bildformate in C

ENDE

Software, die es in sich hat!
Euro Version 3.64

Jetzt auch mit deutscher Anleitung



Nun auch in Deutschland!
Schwarz auf Weiß kann jetzt jeder auf seinem ATARI ST unter MS-DOS arbeiten. Doch nicht nur monochrom, nein, auch in Farbe ist nun der Zugriff auf die Welt der PC-Rechner möglich.
Die Software-Emulation **PC ditto** öffnet allen ATARI ST Anwendern das Tor zum gewohnten professionellen Business Standard.
Mit dem **PC ditto** können Sie mühelos mit Lotus 1-2-3 oder Symphony Ihre Kalkulation erstellen oder Ihre Daten mit DBase III plus verwalten. Mit dem **PC ditto** haben Sie Zugang zu Turbo Pascal, zum GW BASIC Interpreter, und auch zu Borlands neuestem Kind Turbo Basic.
Mit dem **PC ditto** läuft auch Software, die es für den ATARI ST noch gar nicht gibt. Mit dem **PC ditto** laufen so viele Programme, daß wir sie hier gar nicht alle auflisten können.
Der **PC ditto** unterstützt die ATARI-Festplatte, den Druckerport und alle Schnittstellen, soweit es die Hardware zuläßt. Machen Sie aus Ihrem ATARI ST den preiswertesten PC - CLONE! Bestellen Sie den **PC ditto** zum Superpreis von nur DM 198.-.
Wir laden Sie ein! Steigen Sie mit Ihrem ATARI ST in die Welt des MS-DOS ein.



MS-DOS und GW-BASIC sind Warenzeichen von Microsoft Corp. Lotus 1-2-3+ Symphony sind Warenzeichen von Lotus Dev. Corp. Turbo Pascal und Turbo BASIC sind Warenzeichen von Borland Corp. DBase III Plus ist ein Warenzeichen von Ashton-Tate Corp.

Bestellcoupon Merlin Computer GmbH Industriestraße 26 6236 Eschborn Tel.: 06196/481811

Name: ____________
Vorname: ____________
Straße: ____________
Ort: ____________
Unterschrift: ____________

Hiermit bestelle ich:

☐ PC ditto wie oben beschrieben für DM 198.00

☐ Vorauskasse
☐ Nachnahme

Versandkosten: Inland DM 7,50
Ausland DM 10,00
Auslandsbestellungen **nur** gegen Vorauskasse
Nachnahme zuzgl. DM 3,50 Nachnahmegebühr

Kleinanzeigen

BIETE HARDWARE

520ST + SF314 + SM124 + Gehäuse zus. DM 995,–, Farbmon. + Umschalt. günstig. Tel. 07051-20829

SF 354 + 10xDisk 05264/397

★ DIN-A4-Scanner ★ für 1000,– DM zu verkaufen, Tel. 07266/2340

Fabrikneue externe 80MB Amiga 1000 Hard Disk (Phoenix) umständehalber von privat gegen Gebot abzugeben. Tel. 040/73352-0 Herr Hinsch/Herr Berger

---

1 Taylorix-Comp-System 4/W bestehend aus: ZE 112 KB; Diskettenlaufwerk 1,2 MB; DFUE-Einrichtung; Winchesterplatte 14,3 MB; Bildschirm-Tastatur-Drucker DRH 120 Tel. 0671/70630

★★★ Atari ST ★★★
Sammelbestellung
z. B. Mega ST/2 ca. 2300 DM
Axel Schuller, Kleeweg 2
7090 Ellwangen

HARDDISK nach C't-Billiglösung 20–60 MB, Preis auf Anfrage
O. Nothdurft, Schopenhauer 11, 6056 Heusenstamm (06104-5499)

ST1040 + SM124
Tel. 089-7250317

6-achsiger TEACH-ROBOT mit Action-Box DM 750,–
T. 02304/83681

BIETE SOFTWARE

Trimbase 100,–, Steve 200,–, ISGEMDA 80,– Datamat, Text-Design, Textomat zusammen 100,– Timeworks 250,– Tel. 07051-20829

★ ST-PD, Kopie ab 2,– DM ★ alle PD aus ST-Comp. u. eigene Info: T. Helfers, Portslogerstr. 30, 2905 Edewecht · T. 04405/6809

Wordplus, Beckertext, Starwriter ST-Drucktreiber für **alle** Star NL-10 u. LC-10 + Col,LC 24–10, NB 24-10, Kompl. angep., volle Befehlsausnutz.; 1,5 zeil; IBM & Epson; je 30,– DM; DIC 241 K 10 DM; U. Köhler, Mont-Cenis Str. 537 4690 Herne 1; Tel. 02323/64315

Blitter-TOS + altes TOS gleichz. umschaltbar f. alle ST, Anl. K. Ratsch, Herner 127, 4350 Recklh, 02361/28442

●●● Volltreffer! ●●●
PD-Software für ATARI ST und Emulatoren (PC-Ditto/ALADIN) Info f. DM 2,40. Carsten & Marcus Postf. 650602, D-2000 Hamburg 65

■■Public Domain 5,50 DM!!■■
■ immer **aktuelle** PD auf ■
■ Mark.-Disk SS je DM 5,50, ■
■P&V. DM 5,00, ab 20 St. frei!■
■ Achtung!! ■■ Achtung!! ■
■Neue Public Domain auch im■
■„Sommerloch"!! – ART-Lib's■
■für Degas, Stad, Sig., Word+,■
■SIGNUM Fonts/Util. u.v.m.!!■
■ WO? – Schauen Sie ■
■nach der Anzeige an anderer■
■ Stelle in dieser Ausgabe! ■
■ NEUEN Katalog bestellen! ■
■ (für 5,– incl. Probedisk!!) ■
■■ PD-EXPRESS-Versand ■■

Riesenauswahl an PD-Software!
300 verschiedene Disks ab 4,–! Liste gegen –,80 DM Rückporto! Bitte Computertyp angeben!
Die Schnittstelle, Soldinerstr. 14, 1000 Berlin 65, 030/4948820

PD-Grafik-Bilder, Art-Lib für Degas + Stad Katalog: Frey, Rheinstr. 12A, 6538 Münster-Sarmsheim

---

Riesenauswahl an PD-Software!
Diskette ab 2,– DM
Tel. 02721/2432 von 9–22 Uhr

LERNPROGRAMME je DM 29,-
Jägerprüfung: über 400 Fragen Super Tiergrafiken (monochrom)
Erdkunde: Städte-/Länderabfrage BRD/Europa/Amerika/Welt
R. Baumann – 8371 Kirchdorf

ACHTUNG! HANDWERKER!
Preissenkung bei Harosoft HAROFAKT: die ideale Fakturierung für Handwerker. Leistungsbeschreibung mit 8 Zeilen, Kundenverwaltung, Rechnungen, Angebote, L-scheine etc., Mahnwesen, Rundschreiben. Einfach mit der Maus zu steuern!
Bisheriger Preis DM 698,–
Neuer Preis 398,–
Demoversion (+ Kurzanl.) DM 20 HAROSOFT, Fliederstr. 2, 7901 Schnürpflingen, Tel. 07346/3845
ACHTUNG! HANDWERKER!

GROSSE GRAFIKSAMMLUNG ca. 2000 IMG-Grafiken (z. B. für Wordplus) DM 25,– 06302-3338

■ ★ PD-SERVICE ★ ■
■ Einzeldiskette SS **DM 4,00** ■
■ Doppeldiskette DS **DM 6,00** ■
■ (beliebig kombinierbar) ■
■ Porto u. Verp. DM 3,00 ■
■V.-Scheck o. NN (+ DM 3,50)■
■N. Twardoch, Gröchteweg 22■
■ 4902 Bad Salzuflen 1 ■

STAR-TREK-Spiel, Farbe, orig., DM 60,– Tel. 0431-232136 abends

---

Bundesliga Fußball auf dem ATARI ST. Voll GEM unterstützt Tabelle, Spieltag, Analyse, mit Disk 39 DM Thielmann, 02903/2551

Erdkunde lernen mit dem Länder-Hauptstädtequiz M. Kaula Jura-Str. 12, 8602 Altendorf
Vorkasse DM 39,–

ATARI ST P.D. SOFTWARE
Spiele, Anwenderprogramme uvm., **ab 1,70 DM** Gratisliste bei R. Hornung, Salisweg 47, 6450 Hanau 1

Public-Domain-Blitzversand!
Riesenauswahl! Preise:
ab DM 4,– einseitige Disketten
ab DM 6,– doppelseitige
Inclusive Diskette!!!
Auch alle ST-Disks! Auf Doppeldisks beliebig kombinierbar!
Gratisliste anfordern bei:
A. Gauger Software
Buhlstraße 16a, 7505 Ettlingen
0 72 43 / 3 18 28
Bitte Computertyp angeben!!

DATOBERT BUSINESS DM 89,-
Die logische Weiterentwicklung zum PD-Programm **„Datobert"** Demodisk DM 10 / Info g. DM 3,- in Briefm. **Händleranfragen erw.** R. Baumann – 8371 Kirchdorf

---

Lohn-Einkommensteuer, Miet-Lastenzuschuß, Rentenber./Beamtenversorgung
H-I-SOFTWARE, Niederfelderstr. 44, 8072 Manching 08459/1669

■■PD aus „ST" **nur 4,80 DM**■■
■ incl. Markendisk. Ab Nr. 1 ■
■ **24-h-Bestellservice.** ■
■■ Schneider 0281/61772 ■■

★★ Public Domain Software ★★
IBM u. komp. Disk ab 6,– DM
Atari ST Disk ab 3,– DM
Riesenauswahl ★★★ Superpreise
Katalog-Disk für je 1,80 DM
Graf & Schick · Hauptstraße 32 a
8542 Roth · Tel. 0 91 71 / 50 58 - 59

ORIGINALPROGRAMME
Riesenauswahl an neuesten Programmen zu den günstigen – Gauger Software Preisen – ... die Gelegenheit!!
Gratisliste anfordern bei:
A. Gauger Software, Buhlstr. 16a
7505 Ettlingen, 07243/31828

Wir erarbeiten ihre individuelle Lösung für den EDV-Einsatz.
Fertige Programme für:
Hausverwaltung, Techn. Lager, Schiffahrtswesen, Tel. 040/381364

PD-MB: 0571/710141 ★ 300/8/N/1

Wärmebedarfsberechnung DIN4701
KZahlberechnung DIN4108
Dampfdiffusion-Tauwasserschutz
Wärmeschutznachweis WäSchV
Dipl. Ing. V. Koch, Am Mehnacker 11
3563 Dautphetal 3, Tel. 064687652

■■■■ NEC P6-Treiber ■■■
■f. WORD+ f. Ex.-Arb., Dissert.■
■ Prop+Block; 1,5+1zg; ■
■ **Korrekter Seitenumbruch** ■
■■■■ 0203/557729 ■■■■

Wärmebedarf DIN4701 + K-Zahl
★ Heizflächenauslegung ★ Rohrnetz ★ Demodisk 2-seitig für DM 10 Vorkasse von J. Binder, Eichendorffstr. 15 · 5030 Hürth

---

ATARI ST-PD ab DM 2,25, auch f. ALADIN & MS-DOS (PC-Ditto)
INFO: A. Witaseck, Josef-Neuberger-Str. 26, 4 Düsseldorf

Verk. GFA-BA.3.0 02368/55020!

Megamax-Modula2+C neu je 310,–
Eprommer incl. SW Info 08458/9514 – Tannenweg 11 – 80781 Buxheim

SUCHE HARDWARE

Audiodigit. für ST 0 91 26 / 88 40

SUCHE SOFTWARE

Aladin Emulator u. orig. MAC ROMS gesucht. Tel. 02472/7783

Suche Handbuch Easydraw (dtsch) u. Desktop Publisher ST, oder Kopien Tel. 0721/74563

Suche PC-Ditto 3,64, Kauf oder Tausch Tel. 08725-1373

Suche ST-BASE 04331-29273, 18h

PROGRAMMIERER GESUCHT!
T. Zimmermann, Steinacker 12
7520 Bruchsal, Tel. 07251-84170

KONTAKTE

★★ Softwareentwicklung ★★
Schreibe Software, die speziell auf Ihre Bedürfnisse zugeschnitten ist. Ksl. Info anford. Thomas Leufkes, Knappenstr. 10, 4353 Oer-Erkenschwick, 02368/55020

Wer hat Know-how über Atari-CAMBRIDGE LISP!
Tel. 069-893965

Autorenteam schreibt Gebrauchsanweisungen für die von Ihnen entwickelte Software

VERSCHIEDENES

FlugsimulatorII Neuauflage! Deutsche Übersetzung DM 20,– E-Service, Pf 1365, 7312 Kirchheim

---

PD-Software ca. 470 Disketten
Kaufe gebr. Soft- + Hardware
Torsten Duffner, Ritterstr. 6
7833 Endingen, 07642/3875
Katalog kostenlos

K&L Datentechnik

3551 Bad Endbach, Bahnhofstr. 11
Tel.: 02776/202 oder 02776/1030

Lavadraw 3.0: Preis: 99 DM

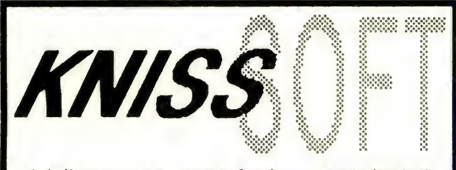
- Professionelles S/W Zeichen- und Bildverarbeitungsprogramm
- mehr als 100 Zeichenfunktionen sind implementiert, alle Standardfunktionen vorhanden, darüber hinaus können Sie z.B. einen beliebigen Ausschnitt (Lassofunktion oder Rechteck): biegen, knicken, verzerren, spiegeln, drehen, verkleinern, vergrößern, auf eine Kugel- oder Zylinderoberfläche projizieren, als Sprühmuster oder Pinsel verwenden und vieles mehr
- erlaubt maßstäbliches Zeichnen: Koordinaten (absolut + relativ) und Lineale können eingeblendet werden, halbautomatisches Bemaßen
- liest + **schreibt** die Formate: SCREEN, DEGAS, IMG, STAD
die Lavadraw-Bilder können in WORDPLUS oder SIGNUM! übernommen werden, weitere Formate in Vorbereitung (Amiga, Mac etc.)
- integrierte SCANNER-Treiber (Handy, HAWK)
- mit dem mitgelieferten Zeichensatzeditor LAVAFONT können eigene Zeichensätze erstellt werden, auch Übernahme der Zeichensätze fremder Programme oder gescannter Vorlagen ist möglich
- Drucker: jeder Matrixdrucker (8-, 16- oder 24-Nadel) kann angepaßt werden Ausdruck längs, quer, 2 Bildschirme nebeneinander, untereinander in wählbarerer Auflösung
DIN A2 Ausdrucke auf NEC P7 oder EPSON LQ 1000 (sieht super aus)
LASER: HP Laserjet + kompatible, ATARI SLM 804
- auch im Paket: SNAP.PRG, speichert Hardcopies aus (fast) jedem Programm auf Diskette -> Weiterverarbeitung mit LAVADRAW

Leider ist der Platz zu klein für eine detailliertere Beschreibung.
Der Bundessoftwareminister: Lavdraw gefährdet ihre Gesundheit und macht süchtig, eine Packung enthält: 1 Programmdiskette, 1 Anleitung und der Preis dafür (abgesehen von der Gesundheit): 99DM

Utility-Disk: Preis: 59 DM

- **Gemdisk V2.0:** resetfeste Ramdisk + Druckerspooler als Accessory
- **Picmaster:** s/w Formatwandler, Screen<->Degas<->IMG<->STAD
WORDPLUS liest IMG-Format, SIGNUM! liest STAD-Format
einfach zu bedienendes GEM Programm
- **4Detect:** Bootsektor-Virusfinder und -killer, voll GEM unterstützt, Anzeige des gesamten Bootsektors, durch einfaches Umbenennen (.ACC statt .PRG) auch als Accessory verwendbar, überwacht auf Wunsch wichtige Systemvariable und gibt Alarm bei Veränderungen
- **Snap:** speichert Hardcopies aus (fast) jedem Programm auf Diskette

Videodata V2.0: Kunden, Film, Ausleihverwaltung für Videotheken incl. Umsatzstatistik **Preis: 249 DM**
VokST: Vokabeln spielend lernen, pro Datei bis zu 2000 Einträge, Fehlerstatistik **Preis: 59 DM**
ODUS LABOR: Abrechnungs- und Verwaltungsprogramm für Dentallabors **Preis: 2199 DM**
Picture Disks: 1 DS-Disk enthält ca. 80 Grafiken **Preis: 20.-**
Format angeben (Stad, Degas, Img usw.) + Diskformat SS o. DS
PUBLIC DOMAIN: Alle ST-PD's **6 DM, 2 PD's auf DS-Disk 9.-**
INFOS anfordern! (2 DM in Briefmarken beilegen)
Versandkosten: Vorkasse (Scheck): **5 DM**, Nachnahme: **7.50 DM**
Händleranfragen erwünscht! * Alle Preise sind unverbindlich empfohlene Verkaufspreise
Wir programmieren auch individuelle Lösungen!



Adalbertstr. 44 - 5100 Aachen - 0241/24252

1st PROPORTIONAL

siehe Testbericht ST-Computer 6/87

1st PROPORTIONAL ermöglicht den Ausdruck von 1st Word Plus Texten in PROPORTIONALSCHRIFT im BLOCKSATZ ! Unterstützt alle proportionalschriftfähigen Nadeldrucker (9, 24 Nadeln) und Typenraddrucker mit PS-Typenrad sowie Kyocera LASER.

- läuft vollständig unter GEM (SW-Monitor)
- unterstützt alle Optionen von 1st Word Plus
- verschiedene Zeilenlineale in Proportionalschrift im Blocksatz
- spezielle Grafiktreiber für 9,18,24 Nadeldrucker, dadurch endliche satte schwärzen auch auf 24 Nadel Druckern sowie unverzerrter Ausdruck
- Verkettung von bis zu 25 Texten mit automatischer Seitennummerierung
- Lieferumfang: Diskette, 60 seitiges deutsches Handbuch
- Druckertreiber für: NEC P2200,P5/6/7, EPSON LQ 500, LQ 800,850,1000,1050,2500, FX 85, SEIKOSHA SL 80 AI, STAR NL 10 m.par.Interface, BROTHER HR 15,20,25,35, TA GABI 9009, KYOCERA LASER F1010,1100,1200...
- 7-seitiges Info mit Probeausdrucken anfordern (2 DM).

1st PROPORTIONAL kostet nur 95 DM

SIGNUM II	369,-	Easy Draw 2/CAD	222,-
Publishing Partner	222,-	ST Paint Plus	119,-
PROTEXT V2.1	127,-	Public Painter	79,-
TEMPUS V2.0	95,-	STAD	154,-
1st PROPORTIONAL	95,-		
		KAISER,das Spiel	114,-
Megamax C	329,-		
Mark Williams C	319,-	fibuMAN e	365,-
Megamax MODULA 2	329,-	fibuMAN f	725,-
Devpac Assembler ST	137,-	fibuMAN m	935,-
		fibuSTAT	398,-
dB MAN	345,-		
Superbase	222,-	Harddisk im ST Gehause	
Superbase PROF	589,-	EX 40 - 40 MB	1795,-
ADIPROG/ADITALK	177,-	EX 60 - 60 MB	2395,-

unverb. GESAMTKATALOG anfordern !

DELO Comp. Tech.

Sommer Preis Hits

Doppelstation D50 2x1MB
bestückt mit 2 NEC 1037A
kompl. anschlußfertig ..... **nur 498.-**

Diskettenstation D26 1MB
3,5" NEC 1037 A A-Laufwerk für ATARI ST
Frontblende wahlweise schwarz oder grau
kompl. anschlußfertig ..... **nur 275.-**

NEC MultiSync GS
inkl. Monitor-Umschaltung ATARI ST an NEC MultiSync alle drei Auflösungen.
Test ST Computer 1.88 ... **nur 588.-**

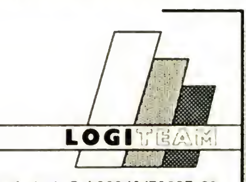
NEC 1037A mit Gehäuse und Einbaumaterial .... **nur 205.-**

☎ 0231/356511



die maus ist raus

mit passender software für alle ataris DM 128,-



Kölner Str. 132 5210 Troisdorf Tel. 02241/71897-98

iks Public Domain

Angebot wie 'ST-Computer'

Einzeldisk SS ab 6 Stück	DM 5.00
bis 5 Stück	DM 6.00
Doppeldisk z.B 1/2...81/82 u.s.w. (aufeinanderfolgend und ungerade beginnend	DM 7.00
Pakete 1-10, 11-20, 21-30, 31-40 u.s.w. auf 5 Disketten DS	DM 30.00

Preis inkl. Disk zzgl. Porto u. Verp. DM 5.00 (Ausl. DM 10.00) NN plus 1.70, besser V.-Scheck

MS DOS Emulator

Haben Sie Probleme Ihre 5,25" MS Dos Software auf das 3,5" Format zu bringen. Wir lösen das für Sie! Wie? Rufen Sie uns an! Preis **DM 15.00** ab 6 St. **DM 12.00**, ab 10 St. **DM 10.00** je Disk

iks
Schönblickstraße 7
7516 Karlsbad 4
☎ **ab 18 Uhr 07202/6793**



Einnahmeüberschußrechnung Finanzbuchhaltung

„BILANZEN MAUSGEKLICKT", ST Magazin 4/88

... Dem Programm bestätigen wir mit gutem Gewissen, daß es absolut fehlerfrei läuft...
fibuMAN setzt ohne Zweifel neue Maßstäbe...

Checkpoint, DATA WELT 3/88

... Beeindruckend sind die umfangreichen Möglichkeiten und die Verarbeitungsgeschwindigkeit...
... braucht sich fibuMAN kaum vor der DATEV zu verstecken...

Weitere positive Tests ST-Computer und c't.

FibuMAN



Schicken Sie mir Infos über uvb. MS-DOS ☐ Atari-ST ☐

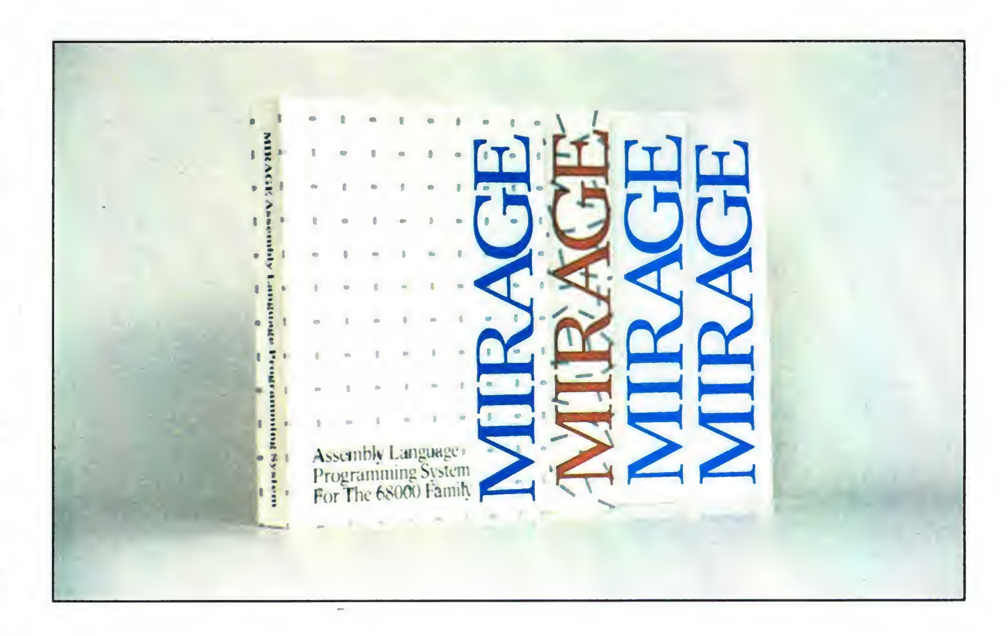
Name, Vorname
Firma
Straße, Hausnummer
Plz, Ort
Telefon

Prodata GmbH · Broicherstr. 39 · 5060 Bergisch Gladbach 1
Telefon (02204) 51456

MIRAGE FÜR DEN ATARI ST

Ein Multiuser-/Multitasking-Betriebssystem

Das in England verbreitete Multiuser-/Multitasking-Betriebssystem MIRAGE ist nun auch für den ATARI ST erhältlich. Ursprünglich für Workstations entworfen, soll es nun den Luxus einer solchen auch für den einfachen Feld-, Wald- und Wiesen-Hacker bringen - zu einem Bruchteil der Kosten. Dieser Test zeigt, ob dies gelungen ist.



Geliefert wird MIRAGE in einem ROM-Modul mit dem eigentlichen Betriebssystem und einigen wichtigen Utilities. Eine Systemdiskette, eine Diskette mit Tastaturtreibern (u.a. für deutsche Umlaute) und zwei Handbücher vervollständigen das Ganze. Das dickere Handbuch ist eine allgemeine Einführung in MIRAGE und für die verschiedensten Computermodelle gültig. Am Anfang wird auf die grundlegenden Eigenschaften von MIRAGE eingegangen, im weiteren werden die Systemutilities in alphabetischer Reihenfolge ausführlich dargestellt. Es folgt die Beschreibung des Editors und schließlich werden die Terminalfunktionen aufgelistet. Das zweite, dünne Handbuch gibt spezielle Informationen zur Implementation auf dem ST. Beide Handbücher sind in gutem, verständlichem Englisch geschrieben, übersichtlich gegliedert und erfüllen ihren Zweck. Allerdings fehlt ein Schlagwortregister, somit ist die Einarbeitungsphase mühsam, da man ständig am Blättern ist. Abhilfe ist aber von der deutschen Vertriebsfirma in Aussicht gestellt: Eine deutsche Übersetzung mit Schlagwortregister befindet sich in Arbeit. Einzig der Systemprogrammierer ist enttäuscht darüber, daß die System-Aufrufe fehlen. Diese werden aber im Handbuch zum ALPS (Assembly Language Programming System) nachgeliefert. Da es - wie später noch genauer erläutert wird - sowieso empfehlenswert ist, sich dieses Softwarepaket zuzulegen, sollte dies nicht als allzu großer Nachteil gewertet werden, was die Dokumentation betrifft. Schließlich kann man ohne Assembler nicht so besonders viel mit den System-Aufrufen anfangen.

Booten unter GEM

Das ROM-Modul ist eingesteckt, die Systemdiskette eingelegt und schon kann es losgehen. Als erstes fragt MIRAGE höflich nach Datum und Uhrzeit - sofern man nicht glücklicher Besitzer eines MEGA-STs ist, bei dem beides automatisch von der Echtzeituhr eingelesen wird. Dann beginnt der eigentliche Bootvorgang, der natürlich - ähnlich wie beim TOS - durch eine AUTOBOOT.CMDF-Datei beeinflußt werden kann. Diese ASCII-Datei wird zeilenweise abgearbeitet, und mit ihrer Hilfe läßt sich das System den persönlichen Bedürfnissen anpassen. In Bild 1 sehen Sie zum Beispiel die AUTOBOOT.CMDF-Datei, die ich mir zusammengestellt habe.

Hat man das Booten glücklich hinter sich gebracht, landet man schließlich in der Shell. Mit GEMialitäten ist es nun vorbei, man hat ein schlichtes Textterminal vor sich. Aber auch hier gibt es einen Lichtschimmer am Horizont: Sahara Soft arbeitet bereits an einer Umsetzung von XWINDOWS für Mirage. Wann diese allerdings abgeschlossen ist, läßt sich noch nicht absehen. Vom Speicherplatzbedarf her scheint MIRAGE zunächst ziemlich genügsam zu sein - der vollständig in Assembler geschriebene MIRAGE-Kern belegt nur ca. 38K. Bei meiner Arbeit mit MIRAGE habe ich aber festgestellt, daß ein ST mit 1 MB freiem Speicher und einem doppelseitigen Diskettenlaufwerk wohl als Minimalkonfiguration gelten muß, wenn die ganzen kleinen Annehmlichkeiten wie RAM-Disk, Drucker-Spooler oder Cache-Speicher eingesetzt werden. Da außerdem viele Systemutilities diskresident sind, ist eine Festplatte

oder zumindest ein zweites Diskettenlaufwerk erstrebenswert. Mit nur einem Laufwerk geht es auch, doch sollte man sich dann auf jede Arbeitsdiskette den Ordner mit den Systemutilities kopieren. Überhaupt muß ich sagen, daß eine Festplatte nicht allein wegen der größeren Kapazität wünschenswert ist, die Geschwindigkeit der Floppy war - offen gestanden - sehr mäßig. Vielleicht bin ich von Fastload verwöhnt, aber unter TOS kann mit der Diskette wesentlich fixer gearbeitet werden.

MIRAGE-Utilities

Unter MIRAGE können Programme und Daten mit dem Kommando 'SHARE' dauerhaft in einem '*shareable memory*' untergebracht werden. Der Zugriff auf solche im Hauptspeicher gehaltenen Programme oder Daten wird natürlich enorm beschleunigt. Ein weiterer Vorteil dieser Maßnahme ist, daß im Multiuserbetrieb solche Programme - sofern sie reentrant (=wiedereintrittsfähig) sind - von mehreren Usern oder Tasks gleichzeitig verwendet werden können, obwohl sie nur einmal im Speicher stehen. Nachteilig an dieser Vorgehensweise ist nur, daß einmal im Speicher verankerte Programme oder Daten dort bis zum nächsten Systemstart bleiben. Eine Möglichkeit, sie mal eben so rauszuwerfen, wenn der Platz eng wird, gibt es nicht. Man sollte sich also gut überlegen, was man wirklich ständig braucht. Besitzer eines MEGA ST können da natürlich nur müde lächeln. Sinnvoll ist es zum Beispiel, den Editor, den Pascal-Compiler und den Macro-Assembler im Hauptspeicher unterzubringen.

Von den über 100 Systemutilities sind 44 in der Shell fest eingebaut. Weitere 20 werden beim Booten aus dem ROM-Modul in das 'shareable memory' transferiert, so daß ungefähr 40 bis 50 von der Diskette oder Harddisk bei Bedarf nachgeladen werden müssen. Von der großen Anzahl an Systemutilities möchte ich nur einige wenigevorstellen: Besondere Bedeutung kommt natürlich dem mitgelieferten Editor zu, stellt er doch quasi das Herz eines jeden Programmentwicklungssystems dar. Zu MIRAGE wird ein Bildschirmeditor mitgeliefert (Bild 2), der in puncto Geschwindigkeit und Funktionsumfang eigentlich keine Wünsche offen läßt. Natürlich darf man keine Tempus-Umgebung mit Fenstern und Maus erwarten, denn, wie schon gesagt, sitzt man vor einem reinen Textterminal.

```
;*******************************************************************;
;*                                                                 *;
;*        AUTOBOOT.CMDF                                            *;
;*        - wird beim Systemstart automatisch aufgerufen           *;
;*                                                                 *;
;*******************************************************************;

; Boltons - Kernelerweiterungen:

  bolton ieeefpak          ; Float-Routinen
  bolton spooler           ; Drucker-Spooler

; Programme/Daten, die im Speicher gehalten werden:

  share edit               ; Full-Screen-Editor
  share pascal.rtp         ; Run-Time-Package für Pascal
  share fortran.rtp        ; Run-Time-Package für Fortran

; Laufwerke:

  device name=dsc1:, driver=atadisc, drive=1; zweites
  Laufwerk
  device name=ram,
  driver=ramdisc, ddvpar='720,9'  ; 360K   RAM-Disk

; Terminal fuer Drucker-Spooler einrichten:

  terminal name=prin1,iobd=atacent    ; neues Terminal
  attach prin1,spoolr              ; Terminal an Spooler
  zuweisen
```

Bild 1: Beispiel einer AUTOBOOT.CMDF-Datei

```
Editing BENCH.PAS        ** Command Mode **
*=======*=======*=======*=======*=======*=======*=======*=======*======
File Size:   2100                              Text not modified
Free Characters: 261018

           D - Delete a Block of Text from Memory
           C - Copy a Block of Text
           M - Move a Block of Text
           W - Write a Block of Text to the Disc
           I - Insert a File from the Disc
           L - List the File Directory on the Screen
           S - Search for a string of characters (Options: F,W,C,N)
           R - Replace a string of characters (Options: F,W,C,N)
           E - Execute MIRAGE command

           Q - Quit without updating the Disc
           F - Finish and update the Disc
           X - Execute Extension Specific Compiler (Option: P)

           H - Help

Command .
```

Bild 2: Menü des Editors

```
-pascal bench.pas
MIRAGE MC68000 Pascal Compiler Version 3.4.31
Copyright (c) 1987 Swifte Computer Systems Ltd.
Compiling BENCH
Phase 1 - Syntax Checking - Workspace 206128 bytes
Source lines - 122 ; User symbols - 27 ; Space used - 4976 bytes
Phase 2 - Code Generation - Workspace 201168 bytes
Successful compilation of BENCH
MIRAGE MC68020 Optimising Macro Assembler Version 2.0
Copyright (c) 1987 Swifte Computer Systems Ltd.
 99  Symbols Defined     0  Macros Defined
Program File BENCH.PROG Completed
-_
```

Bild 3: Aufruf des PASCAL-Compilers
Das Programm BENCH.PROG liegt zum Schluß fertig auf der Diskette vor

Benchmarks

Die im folgenden vorgestellten Benchmarks habe ich dem c't-Computermagazin 10/1987 entnommen. Damit wollte ich mir keineswegs nur Arbeit ersparen - eine Schleife, in der 1000mal irgendwas berechnet wird, könnte ich zur Not auch noch selber fabrizieren. Für den Leser bringt es aber einen großen Vorteil, wenn verschiedene Computer-Blätter dieselben Benchmarks verwenden. So kann man ohne weiteres die Leistung verschiedener Computer miteinander vergleichen, die in den unterschiedlichen Publikationen getestet wurden.

Die Benchmarks sind in PASCAL geschrieben, lassen sich aber leicht in gängige Sprachen (Fortran, Basic, C, Pearl, ...) übertragen. Es muß nur ein Timer vorhanden sein, um die Laufzeit messen zu können.

```
program HLbenchmark (input,output);
   var       h,m,s,ms,hold,mold,sold,msold: integer;
procedure inthmath;
      var
      i,x,y: integer;
   begin
      time (hold,mold,sold,msold); * spezifischer Befehl um die Zeit abzufragen *
      x:= 0;
      y:= 3;
      writeln ('start');
      for i := 1 to 10000 do
      begin
         x:= x+(y*y-y)div y
      end;
      writeln ('finish',x);
      time (h,m,s,ms);
      writeln (hold,mold,sold,msold);
      writeln (h,m,s,ms);
   end;
procedure realmath;
   var
      i:   integer;
      x,y: single;             * Fließkomma - einfache Genauigkeit *
   begin
      time (hold,mold,sold,msold);
      x:= 0.0;
      y:= 9.9;
      writeln ('start');
      for i:= 1 to 10000 do
      begin
         x:= x+(y*y-y)/y;
      end;
      writeln ('finish',x);
      time (h,m,s,ms);         writeln (hold,mold,sold,msold);
      writeln (h,m,s,ms);
   end;
procedure triglog;
   var
      i:   integer;
      x,y: single;
   begin
      time (hold,mold,sold,msold);
      x:= 0.0;
      y:= 9.9;
      writeln ('start');
      for i:= 1 to 1000 do
      begin
         x:= x+sin(arctan(cos(ln(y))));
      end;
      writeln ('finish',x);
      time (h,m,s,ms);
      writeln (hold,mold,sold,msold);
      writeln (h,m,s,ms);
   end;
procedure store;
   var
      i:          integer;
      y:          string[20];
      testfile:   text;
   begin
      time (hold,mold,sold,msold);
      writeln ('start');
      y:= '1234567890qwertyuiop';
      assign (testfile, 'hltest.test');
      rewrite (testfile);
      for i:= 1 to 1000 do
      begin
```

Listing geht weiter...

Aber der erfahrene Programmentwickler benutzt sowieso meistens CONTROL-Sequenzen, so daß man mit diesem Editor nach einer kurzen Einarbeitungsphase gut zurechtkommen dürfte. Vorteilhaft in diesem Sinne ist auch, daß man die Tastenbelegung beliebig ändern kann, zum Beispiel in Richtung Wordstar.

Mit FILECOMP und SEARCH existieren Mini-Textfilter, die es ermöglichen, zwei Dateien zu vergleichen oder eine bestimmte Zeichenkombination zu suchen. Mit LINES läßt sich die Zeilenanzahl einer Text-Datei feststellen. Anzumerken ist noch, daß die Programmiersprachen Pascal, Fortran, Macro-Assembler und Basic ebenfalls wie Systemutilities behandelt werden. Man ruft sie also einfach mit dem Namen auf (Bild 3).

Boltons

Boltons sind sozusagen Kernel-Erweiterungen, die es dem Systemprogrammierer erlauben, das System um eigene System-Aufrufe zu erweitern. Einige Teile der MIRAGE-Software sind als Boltons ausgelegt, um nicht unnötig Speicher zu verschwenden. So gibt es zum Beispiel ein Bolton für die Fließkomma-Routinen nach IEEE 754, die von allen Swifte-Compilern (Pascal, Fortran, Basic und Lisp) genutzt werden. Das hat den Vorteil, daß beim Einsatz eines Arithmetik-Chips (MC68881 oder NS16081) die Programme nicht mehr neu kompiliert werden müssen. Man lädt einfach das entsprechende Bolton und los geht's. Weiter gibt es Boltons für einen Druckerspooler und zwei verschiedene Cache-Programme.

Der Druckerspooler ist in der Lage, bis zu 50 Dateien in eine Warteschlange einzureihen und auf 30 verschiedene Drucker zu verteilen. Diese Fähigkeit wird man zwar auf einem ATARI ST nicht so ganz ausschöpfen können, doch kann man bereits ohne große Probleme drei verschiedene Drucker anschließen (einen Laser am DMA-Port, einen Drucker an der seriellen und einen an der parallelen Schnittstelle), die dann bequem mit Arbeit versorgt werden können.

Die beiden Cache-Programme verfolgen zwei verschiedene Philosophien. DISCACHE versucht, die am häufigsten gebrauchten Sektoren zu puffern, und arbeitet bereits bei einer Puffergröße von 32K.

TRACACHE dagegen puffert immer einen ganzen Track, denn es geht davon

aus, daß der nächste benötigte Sektor sich auf demselben Track befindet, und die Einlesezeiten für einen ganzen Track nicht viel größer sind als für einen speziellen Sektor. Allerdings benötigt TRACACHE als Minimum 256K, schluckt also einen ganz schönen Brocken von meinem kärglichen Megabyte. Am besten ist es, wenn Sie durch Probieren das für Ihre Anwendungen am besten geeignete Cache-Programm herausfinden.

Nieder mit der Hierarchie

Gewöhnungsbedürftig ist das MIRAGE-Dateisystem. Wer sich bei TOS und MS-DOS an das so schön übersichtliche hierarchische System gewöhnt hat, ist erst einmal etwas verwundert: Die Dateien werden von MIRAGE in gleichrangigen Directories verwaltet, es gibt kein Wurzelverzeichnis. Das fehlende Wurzelverzeichnis hat mich besonders gestört, denn formatiert man beispielsweise eine Diskette, so kann man nicht einfach anfangen und seine Dateien darauf abspeichern. Vorher ist es notwendig, ein Verzeichnis anzulegen, in welches dann die Dateien kopiert werden können. Das gilt auch für die RAM-Disk. Das COPY-Utility nimmt einem aber einen Teil dieser Arbeit ab.

Unter MIRAGE verarbeitet der ST drei verschiedene Diskettenformate: 360 KByte (einseitig), 720 KByte (doppelseitig) und 800 KByte (doppelseitig, zehn Sektoren pro Track). Zuweilen hat man nun das Bedürfnis, die eingelegte Diskette gegen eine andere auszuwechseln. Gewöhnlich drücke ich hierzu den Auswerfknopf an meinem NEC 1036A - mit dem ich übrigens sehr zufrieden bin (gelle, Claus!) -, und wenn nicht gerade der Schutzschieber wie ein Enterhaken verbogen ist (alles schon vorgekommen!), kommt die Diskette auch anstandslos heraus. So einfach ist die Sache bei MIRAGE nun nicht. Als ordentlicher Mensch muß man die eingelegte Diskette erst einmal abmelden (UNMOUNT FLP0). Nun kann man die Diskette ohne Gefahr für Leib und Leben entnehmen und die neue einlegen. Diese muß natürlich wieder angemeldet werden (MOUNT FLP0). Vergißt man diese Prozedur, droht das Handbuch den Verlust von Dateien an.

In der Praxis wird das An- und Abmelden zum Glück nicht so streng ausgelegt, doch wenn Sie - wie ich - mit verschiedenen Diskettenformaten arbeiten wollen, ist

```
         write (testfile,y);
      end;
      close (testfile);
      erase (testfile);
      writeln ('finish');
      time (h,m,s,ms);
      writeln (hold,mold,sold,msold);
      writeln (h,m,s,ms);
   end;
procedure textscreen;
   var
      i:  integer;
   begin
      time (hold,mold,sold,msold);
      writeln ('start');
      for i:= 1 to 1000 do
      begin
         writeln ('1234567890qwertyuiop',i);
      end;
      wrtieln ('finish');
      time (h,m,s,ms);
      writeln (hold,mold,sold,msold);
      writeln (h,m,s,ms);
   end;
procedure graphscreen;      * nur auf Grafik-Terminals sinnvoll *
   var
      x,y: integer;
   begin
      time (hold,mold,sold,msold);
      for x:= 1 to 100 do
         for y:= 1 to 100 do
            plot (x,y);      * spezieller Befehl zum Punktsetzen  *
         end;
      end;
      writeln ('finish');
      time (h,m,s,ms);
      writeln (hold,mold,sold,msold);
      writeln (h,m,s,ms);
   end;
begin
   textscreen;
   intmath;
   realmath;
   triglog;
   store;       * eine freie, frisch formatierte Diskette sollte *
                * eingelegt sein                                  *
   graphscreen; * nur möglich, wenn Betriebssystem Grafik         *
                * unterstützt                                     *
end.
```

Listing Ende

ein MOUNT FLP0 nach einem Diskettenwechsel anzura-ten. Ärger kann es trotzdem geben. So konnte ich, nachdem ich mit einer auf 800 KByte formatierten Diskette gearbeitet hatte, trotz der ganzen MOUNT-Prozedur einfach nicht auf eine spezielle Diskette - die das ALPS-Programmpaket enthielt - zugreifen.

Nach einem neuen Systemstart ging es dagegen ohne weiteres, doch wehe, ich legte zwischendurch mal wieder eine 800-KByte-Disk ein. Wenn Sie eines der beiden Cache-Programme einsetzen wollen, ist das An- und Abmelden natürlich dringend anzuraten, denn sonst kann es passieren, daß nach einem Wechsel der alte Cache-Inhalt auf die neue Diskette geschrieben wird. MIRAGE bietet zwar einen Schutzmechanismus: Es vergleicht bei jedem Dateizugriff eine Checksumme auf der Diskette mit einer im Speicher. Ist diese nicht identisch, so meldet MIRAGE automatisch die neu eingelegte Diskette an. Doch 100%ig ist dieser Schutz halt doch nicht. Fairerweise muß man zu dieser ganzen An- und Abmelderei sagen, daß es sich bei MIRAGE ja um ein Multiuser-/Multitasking-Betriebssystem handelt, und daß man in einem solchen nicht einfach mal eben eine Diskette rausnehmen kann. Haben andere Jobs oder Tasks zum Beispiel noch geöffnete Dateien auf dem Gerät, welches Sie UNMOUNTEN möchten, so wird dies vom Betriebssystem entrüstet - und das zu Recht - zurückgewiesen.

Im Harddisktreiber meiner Version ist übrigens noch ein Fehler zu finden. Auf Festplatten, die unter TOS bootfähig sind, läßt sich MIRAGE nicht installieren. Das ist weiter nicht tragisch - eher unangenehm und mit Arbeit verbunden, denn es ist nötig, ein Backup von seiner Platte zu machen und sie anschließend neu - aber diesmal ohne Autoboot - zu partitionieren. Nach Rücksprache mit Claus Brod, dem Floppy- und Harddisk-Spezialisten, gibt es allerdings doch ein Mittel namens SED, um eine bootfähige Platte sozusagen zu entmannen. Näheres steht im

Benchmark-Ergebnisse:

	MIRAGE/Pascal	RTOS/Pearl
Textscreen:	55,080	31,217
Intmath:	0,765	0,445
Realmath:	5,850	3,064
Triglog:	4,735	4,544
Store:	21,000	6,349

Zeit in Sekunden, Millisekunden

MIRAGE wurde im Single-User-Modus betrieben, das heißt, es war nur ein Job deklariert. Natürlich lief bei beiden Betriebssystemen jeweils nur das Task mit den Benchmarks.

Die Zeiten für MIRAGE sehen bei Textscreen und Store nicht so besonders gut aus. Bei Store liegt MIRAGE im Vergleich zu TOS-Sprachen (GFA-Basic, ST-Pascal-Plus, ...) eigentlich gut im Rennen. RTOS verwendet für das Diskettenhandling neue, eigene Routinen, deswegen ist es so schnell. Die Zeiten für Textscreen würde ich dem Pascal-Compiler anlasten, denn der MIRAGE-Editor ist beim Textscrollen doch recht schnell.

SCHEIBENKLEISTER, wo derlei Perversitäten ausführlich beschrieben werden.

Ein kurzes GFA-BASIC-Programm sollte obiges Problem ohne viel Pipapo lösen - so dachten wir. Nach einer längeren Sitzung vor dem Rechner mußten wir feststellen, daß MIRAGE sich hartnäckig weigerte, auf eine Platte installiert zu werden, auch wenn sie mit dem erwähnten Programm bearbeitet worden war. Irgendwie muß sich MIRAGE um Bytes auf der Platte kümmern, die es eigentlich gar nichts angehen, denn unter TOS funktionierte das Programm tadellos, die Platte war nicht mehr bootfähig. Schließlich gaben wir auf, machten ein Backup, partitionierten die Platte neu und probierten es dann. So einfach wie im Handbuch beschrieben ließ sich MIRAGE allerdings nicht auf der Platte installieren. Nach mehreren Anläufen haben wir es dann doch noch geschafft. Das Problem war, den TOS-Treiber AHDI.PRG zusammen mit MIRAGE in den Speicher zu kriegen. Wir haben es folgendermaßen gelöst: Auf einer TOS-Diskette einen AUTO-Ordner einrichten, den AHDI zuerst hineinkopieren, dann MIRBOOT.PRG, um MIRAGE zu starten. MIRAGE meldet sich dann nach kurzer Zeit und meckert an, daß keine MIRAGE-Diskette im Laufwerk liegt. Nichts einfacher als das, man legt halt die MIRAGE-Support-Disk ein, und anschließend wird anstandslos gebootet. Hört sich etwas umständlich an - und ist es auch. Der Versuch, AHDI.PRG vom Desktop aus zu starten und danach MIRBOOT.PRG, ist kläglich gescheitert. Im nächsten Update von MIRAGE sind die Probleme, die bei einer bootfähigen Platte auftreten, behoben.

Mehr Schein als Sein

Wer kennt nicht das Problem, daß ehemals harmlose Adreßdateien von lieben Freunden, Bekannten und den leidigen Anverwandten plötzlich zu riesigen, grauenvollen Monstern anschwellen, die sich gierig auf jedes hilflose Byte stürzen, um es zu verschlingen? Zaubern kann MIRAGE zwar nicht, doch bietet es eine feine Möglichkeit, obiges Problem hinauszuschieben, nach dem Motto: Was Du heute kannst besorgen - Mensch, das kannst Du auch noch morgen.

Oft kommt es vor, daß der Platz auf der Platte zwar prinzipiell reicht, aber falsch verteilt ist. So möchte man eine 7 MByte große Datei unterbringen, hat aber dummerweise nur zwei Partitionen mit je 5 MByte frei. In diesem speziellen Fall kann man sich mit MIRAGE elegant aus der Affäre ziehen - wie weiland Münchhausen am eigenen Schopfe: Zwei oder mehr physikalische Laufwerke werden zu einem virtuellen Laufwerk zusammengefaßt. Das nennt sich dann SUPERVOLUME und funktioniert auch mit Floppylaufwerken. Ein Programm, welches die Daten verwaltet, merkt von alldem natürlich nichts. Durch diese Möglichkeit erspart man sich unter Umständen zeitraubende Umstrukturierungen seiner Festplatte.

Wanderer zwischen den Welten

MIRAGE erhebt Anspruch darauf, als Entwicklungssystem für TOS eingesetzt zu werden. Eine unabdingbare Vorraussetzung hierfür ist, daß Dateien zwischen beiden Betriebssystemen ausgetauscht werden können. Nun existiert MIRAGE schon seit mehr als 6 Jahren und in dieser Zeit hat es seine Flexibilität mehr als einmal unter Beweis stellen müssen. So verwundert es nicht, daß Treiber existieren, um TOS-, MS-DOS- und UCSD-p-Dateien zu lesen. Ein weiteres Programm ermöglicht das Beschreiben von TOS-Disketten. Damit ist die Bahn für MIRAGE frei, um in fremden Betten zu wildern.

1, 2, 3, viele

Im Zuge der MEGA-Euphorie hat sich vielleicht der eine oder andere einen zweiten ST angeschafft und weiß nun nicht so recht, was er Sinnvolles mit seinem altem Gerät anstellen soll. MIRAGE bietet zwei Möglichkeiten, seinen Oldie zu recyclen:

Einmal kann ein zweiter ST als Terminal über die serielle Schnittstelle betrieben werden. Das ist eine feine Sache bei der Programmentwicklung, denn nun kann man sein Programm debuggen, ohne eine eventuell vorhandene Bildschirmmaske zu zerstören.

Da MIRAGE allerdings auch virtuelle Terminals bietet, reicht allein für diesen Zweck auch schon ein ST. Die Möglichkeit, ein zweites physikalisches Terminal anzuschließen, ist aber auch interessant, wenn zum Beispiel die Freundin oder der Freund unter MIRAGE arbeiten wollen, selber aber dasselbe nicht besitzen. Man ist dabei natürlich nicht an einen ST als Terminal gebunden, sondern jeder andere Computer, der über eine Terminalemulation verfügt, kann angeschlossen werden, notfalls sogar über das Telefonnetz.

Die zweite Möglichkeit ist schon etwas aufwendiger. MIRAGE ist netzwerkfä-



UVS-Software

Händler-Anfragen erwünscht!
Ihr Lern-Software-Spezialist!

Atari Special Nr. 4/88 (S. 29) berichtet auf 2½ Seiten über Lern ST und endet mit dem Fazit: **„Lern ST ist ein ideales Fremdsprachen-Lernprogramm, das auch dem Computer-Einsteiger kaum Probleme bereiten dürfte . . .“.**

Lern ST-Universal-Lernprogramm

GEM-Programm zum komfortablen Üben von Vokabeln aller Sprachen sowie Geschichtsdaten, Biologie . . . Zahlreiche Abfrage-Modi, Statistik, Lexikon, variables Listendrucken, Updates, 160 seitiges Handbuch. **DM 59,–**

Vokabel-Disketten zu Lern ST:

Lern ST-Latein: ca. 3000 Vokabeln und Wendungen, Verben mit Stammreihe, Substant. mit Gen.-Formen usw. **DM 20,–**
Lern ST-Englisch:
ca. 2200 engl. Vokabeln **DM 20,–**
Lern ST-Französisch:
ca. 5000 Vok. und Wendungen **DM 20,–**
Lern ST-Spanisch:
ca. 5000 Vok. und Wendungen **DM 20,–**
Lern ST-Begriffe:
1500 aus verschiedenen Gebieten **DM 20,–**

Möchten Sie lieber **SPIELEND LERNEN?**

Lern-Action-Spiel

Mit diesem interessanten Spiel können Sie Vokabeln aller Sprachen üben. Das Lern-Action-Spiel ist Lern ST-kompatibel, wird mit Englisch-Wörtern und ausführlicher Anleitung geliefert. **DM 49,–**

Für geplagte Latein-Schüler bieten wir ein zusätzliches, spezielles Programm an:

Deklinat-Konjugat

Lateinische Verben konjugieren, Substantive deklinieren, Formen suchen, bilden und abfragen . . .

Dieses ideale **Latein**-Lernprogramm wird mit umfangreichem Wortschatz und Anleitung geliefert. Es ist Lern ST-kompatibel.
Deklinat-Konjugat **DM 79,–**

Ulrich Veigel Softwareservice · Mönchseestr. 83-85 · 7100 Heilbronn · Tel. 07131/60023
Bitte fordern Sie ausführlichen Lern-Softwarekatalog an!

... und dann einen Schritt weiter mit 2nd_Word

Nur DM 59.-



2nd_Word ist ein Textverarbeitungsprogramm, das auf ungewöhnliche Art ungewöhnliche Leistungen bringt, denn 2nd_Word beherrscht Proportionalschrift und Blocksatz und arbeitet nach dem WYSIWYG-Prinzip, d.h. alles, was auf dem Bildschirm bearbeitet wurde, wird genauso auf dem Drucker wiedergegeben.
2nd_Word macht es möglich, eine komplette DIN A4-Seite auf dem Monitor abzubilden (rein softwaremäßig, also nur Bildschirm drehen).
Natürlich können Sie mit 2nd_Word Ihre Texte auch direkt schreiben. 2nd_Word ist ein eigenständiges Textprogramm für alle ATARI ST-Rechner mit monochromen Bildschirm und beinhaltet alle wichtigen Funktionen, die man zum Edieren braucht.

Features:
- Blocksatz und Proportionalschrift auf Bildschirm und Drucker
- Ganzseitenlayout DIN A4 hoch
- WYSIWYG
- leichtes Umformatieren von Texten per Tastendruck
- Großbuchstaben in doppelter Höhe und Breite (Bildschirm und Drucker)
- 1st_Word-Texte können gelesen und geschrieben werden. Übernahme aller Attribute.
- variabler Zeilenabstand
- Tastenprogrammierung (jede Taste)
- Macrodefinition
- weiche Trennung
- Druck mit 8/9- und 24-Nadel-Drucker
- eigene Zeichensätze
- Zeichensatzeditor (für Bildschirm- und Druckerzeichensätze)
- Funktionstastenbelegung (Fett, Unterstrichen, Kursiv, Groß, Unproportional, Sub-/Superscript, Zentrieren, Einrücken, Reformat)

2nd_Word - das eigenständige Textverarbeitungsprogramm oder als Ergänzung zu 1st_Word.



Bestellcoupon Merlin Computer GmbH Industriestraße 26 6236 Eschborn Tel.: 06196/481811

Name: ____________________
Vorname: ____________________
Straße: ____________________
Ort: __________ __________
Unterschrift: ____________________

Hiermit bestelle ich:

☐ 2nd_Word wie oben beschrieben Nur DM 59,00

☐ Vorauskasse
☐ Nachnahme

Versandkosten: Inland DM 7,50
Ausland DM 10,00
Auslandsbestellungen **nur** gegen Vorauskasse.
Nachnahme zuzgl. DM 3,50 Nachnahmegebühr

hig, das heißt, mehrere MIRAGE-STs können vernetzt werden. Beim ST sind dazu nur die Treibersoftware MICROGRID und entsprechende Kabel für die RS-232 oder die MIDI-Schnittstelle notwendig. Benutzt man den seriellen Port, ist man bei der Anzahl der Netzknoten auf zwei beschränkt. Über MIDI lassen sich dagegen Ringe mit vielen STs aufbauen. MICROGRID benutzt dazu bei der Datenübertragung ein umfangreiches Protokoll.

Das setzt natürlich die effektive Datenübertragungsrate erheblich herab, so daß es mir wenig sinnvoll erscheint, dieses Low-Cost-Netz dazu zu benutzen, viele STs ohne Massenspeicher mit einem ST als Datei-Server zu vernetzen. Für diesen Zweck gibt es leistungsfähigere Netze wie Ethernet oder RS-432, wofür dann allerdings auch zusätzliche Hardware nötig wird. Das MIDI-Netz würde ich einsetzen, um mehrere STs mit eigenen Massenspeichern zu koppeln, zum Beispiel für Schulungszwecke an Schulen oder Unis. Interessant ist diese Möglichkeit auch für die Programmentwicklung in Teamarbeit, wo dann das lästige Diskettenaustauschen entfällt. Da MIRAGE umfangreiche File-Security-Möglichkeiten bereithält, ist sichergestellt, daß jeder ungestört arbeiten kann. Bei diesen beiden Anwendungen kommt es nicht so sehr auf die Datenübertragungsrate an, da ein Austausch von langen Dateien nicht allzu häufig vorkommt. Dank Multitasking muß sowieso nur der Empfänger auf seine Daten warten. Wichtiger sind hier Zuverlässigkeit und Unkompliziertheit.

Secret Service

Was Du nicht willst, das man Dir tu', dem schiebe einen Riegel vor. In diesem Sinne verfährt MIRAGE, wenn mehrere Nutzer an einem Rechner oder an einem Netz arbeiten. So gibt es das gute alte Paßwort beim Einloggen in das System, verschiedene Privilegien für einzelne Benutzer, und schließlich kann jeder Eigentümer von 'Resourcen' (z.B. Floppies, Verzeichnissen, Dateien oder Terminals) diese gegen den Zugriff anderer Benutzer schützen. Einzig der 'Superuser' schert sich einen feuchten Kehricht um all das - aber wer glaubt schon an die User?

MIRAGE - oder eigentlich der System-Manager - trägt nun jeden Benutzer mit folgenden Attributen im 'User Authorisation File' ein:

1. Benutzername zum Einloggen (1 bis 6 Zeichen)
2. User-Identification-Code (UIC)
3. Paßwort zum Einloggen (0 bis 8 Zeichen)
4. Heimat-Gerät/Directory
5. Grundeinstellung für die Zugriffserlaubnis (Default-Access-Permission)
6. Spezielle Benutzer-Privilegien
7. CPU-Zeit (Scheduling-Priorität)
8. Speicherplatz, der dem Benutzer beim Einloggen fest zugewiesen wird

Punkt 1 braucht wohl keine nähere Erläuterung, Punkt 2 dagegen ist schon interessanter: Der User-Identification-Code

Technische Daten

Beim Autokauf sind sie für viele das Wichtigste, und auch bei einem Betriebssystem geben sie Anhaltspunkte, die kaufentscheidend sein können. Nachdem ich die Benchmarks unter MIRAGE habe laufen lassen, war ich doch etwas enttäuscht darüber, daß es im Vergleich zu RTOS immer mehr oder weniger langsamer war. Die Werbung verspricht aber ein besonders schnelles Multitasking-/Multiuser-Betriebssystem. Eine Rücksprache mit gdat hat ergeben, daß es von SAHARA SOFTWARE ein Informationsblatt mit den Laufzeiten für Systemaufrufe gibt. In diesem ist zwar ein 68000-Rechner mit 12 MHz getestet worden, aber da dieser Rechner - wie der ATARI ST - mit 0 Waitstates beim Zugriff auf den Speicher betrieben wird, lassen sich diese Zeiten doch ganz gut auf einen ST übertragen. Man muß nur die angegeben Zeiten mit 1.5 multiplizieren, um die Werte für einen 8 MHz-Rechner zu bekommen. Ob diese errechneten Werte nun hundertprozentig mit auf einem ST gemessenen übereinstimmen, läßt sich natürlich nicht mit absoluter Sicherheit sagen. Von den über 60 Zeitangaben habe ich mir einige wenige herausgegriffen, die mir besonders interessant erschienen. Wer die restlichen wissen möchte, kann sich an gdat wenden.

Im folgenden sind die angegeben Zeiten schon mit 1.5 multipliziert, um einen Eindruck von der Leistungsfähigkeit von MIRAGE auf einem ATARI ST zu bekommen.

Taskwechselzeit:	217.5 µs
(Diese Zeit kann sich noch erheblich verkürzen, zum Beispiel wenn eine Task freiwillig einen Wechsel herbeiführt, dauert es nur 149 µs)	
Semaphorzyklus:	
P + V Operation auf Semaphore:	30.3 µs
(entspricht RTOS Request und Release)	
Reaktionszeit auf Interrupt:	63.5 µs
(klinkt man sich direkt in die Interruptvektoren ein, dann verkürzt sich diese Zeit auf 20.5 µs!)	
Block-Kopier-Befehle:	
BLKMOVE (10 KBytes):	6843.2 µs
(dieser Wert ist sehr gut!)	
Um einen 10KByte großen Speicherbereich mit Nullen zu füllen braucht MIRAGE nur	4403.3 µs

Alle Zeiten sind in Mikrosekunden angegeben.

Vergleicht man nun RTOS und MIRAGE, dann stellt man fest, daß MIRAGE verdammt schnell ist - und das, obwohl es nicht speziell für Echzeitbetrieb ausgelegt ist. Die Taskwechselzeit bei RTOS beträgt 200 µs, ist also nicht viel schneller, als die von MIRAGE. Der Semaphorzyklus ist bei beiden Betriebssystemen gleich schnell. Andere Zeiten konnte ich leider nicht vergleichen. Interessant wären noch die Fließkomma-Routinen gewesen, die bei MIRAGE besonders schnell sein sollen.

Das schlechtere Abschneiden von MIRAGE gegenüber RTOS bei den Benchmarks muß also an den Compilern liegen. Auf Betriebssystemebene sind beide einander ebenbürtig.

(UIC) besteht aus zwei Teilen, jeweils einer Zahl zwischen 0 und 255. Der erste Teil des UICs ist die Gruppen-Nummer und weist den Benutzer einer der 256 möglichen Gruppen zu. Der zweite Teil des UICs ist die Mitglieds-Nummer.

Sie unterscheidet die einzelnen Mitglieder einer Gruppe voneinander. Der UIC ist - zusammen mit einer Zugriffserlaubnis - Bestandteil jeder 'Resource' (Gerät, File, Directory, ...) und ermöglicht es so, daß jeder Benutzer seine Resourcen für andere beliebig sperren oder öffnen kann.

Bei Punkt 3 ist lediglich zu erwähnen, daß ein Paßwort kein Zwang ist, sondern optional. Der Benutzer kann sein Paßwort jederzeit mit dem PASSWORD-Utility ändern.

Im Punkt 4 wird festgelegt, wo sich der Benutzer nach dem erfolgreichen Einloggen wiederfindet. Enthält das Directory ein Kommando-File namens SIGNON.CMDF, so wird dies ausgeführt. Es kann zum Beispiel automatisch ein bestimmtes Programm starten.

Die Zugriffserlaubnis für seine Resourcen kann der Benutzer explizit angeben, oder es wird die Grundeinstellung aus Punkt 5 übernommen. Die Zugriffs-Regelung geschieht mit Hilfe eines Access-Permission-Codes (kurz APC), der folgendermaßen aufgebaut ist:

<NSGO>

N - Netzwerk-Zugriff
S - System-Zugriff
G - Gruppen-Zugriff
O - Eigentümer-Zugriff

Jedes dieser vier Zugriffsfelder kann nun wiederum vier verschiedene Werte annehmen:

W - schreiben/löschen/lesen/ausführen
R - lesen/ausführen
E - ausführen
N - kein Zugriff

Einige Beispiele:

<wwww> Alle Benutzer können schreiben, löschen, lesen und ausführen

<nnrw> Gruppenmitglieder können lesen, Eigentümer kann schreiben und lesen

<neew> Kein Zugriff für Benutzer, die über ein Netzwerk angeschlossen sind. Alle Benutzer desselben Systems können ausführen, der Eigentümer darf wieder schreiben und lesen.

<nnnn> Niemand hat ein Zugriffsrecht - man kann ein File also auch vor sich selber schützen.

Nun zu den verschiedenen Privilegien, die unter Punkt 6 angesprochen worden sind:

PHR Physikalisches Lesen ist erlaubt

PHW Physikalisches Schreiben ist erlaubt

SUP Wechselt zum Supervisor-Mode der 68000-Familie. Für einige MIRAGE - Systemaufrufe und Utilities nötig.

OPR System - Operator - Privileg: hat Zugriff auf eine Anzahl von Diagnose - und Konfigurations programmen, aber noch nicht den uneingeschränkten Zugriff eines System-Managers.

MAN System-Manager: hat uneingeschränkten Zugriff auf alle System-Resourcen.

GL Gruppenleiter: hat freien Zugang zu allen Resourcen einer Gruppe

Zu Punkt 7 ist an dieser Stelle nichts weiter zu sagen, auf das Tasking unter MIRAGE wird noch eingegangen.

Punkt 8 ist auch ziemlich klar, dem Benutzer wird beim Einloggen ein fester Speicherumfang zugeordnet, über den er allein verfügen kann. Benötigt er mehr, so wird - sofern noch vorhanden - aus dem freien Speicher-Pool welcher angefordert. Beendet ein Benutzer seine Sitzung, wird der von ihm belegte Speicher an den freien Speicher-Pool zurückgegeben.

Der aufmerksame Leser fragt sich nun: Wie ändert ein nicht privilegierter Benutzer sein Paßwort? Denn um dieses zu ändern, muß die USERS.AUTH-Datei geändert werden, auf die der normale Benutzer aber keinen Zugriff hat. Andererseits habe ich schon gesagt, daß dies mit dem PASSWORD-Utility ohne weiteres möglich ist. Nun, der Eigentümer von PASSWORD ist der System-Manager, welcher uneingeschränkten Zugriff auf das ganze System hat. Mit dem ASSUME-OWNER-I.D.-Attribut überträgt man die Privilegien des *Eigentümers* auf das Programm. Jetzt ist auch der gemeine Benutzer in der Lage, sein Paßwort jederzeit zu ändern. Um das ASSUME-OWNER-I.D.-Attribut zu setzen, genügt es, die spitzen Klammern um den ACCESS-PERMISSION-CODE umzudrehen: >eeew<

Alles schön und gut, denkt sich da der normale ST-Anwender, aber ich werde bestimmt nicht Dutzende Terminals an meinen Rechner anschließen, was bringt's mir also? Ich sehe da im großen und ganzen zwei Vorteile: Zum einen ist der Paßwort-Mechanismus ganz nützlich, wenn man verhindern will, daß jeder Da-

```
Debuging MYPROG.PROG                    MYPROG.PROG  Base=E9242:  Size=1008
A0: 000E8D00 A1: 00000000 A2: 00000000 A3: 00000000  PC: 000E9242:1:000000
A4: 00000000 A5: 00000000 A6: 000E9242 A7: 00069062  SR: 0000      ..0.....
D0: 00000000 D1: 00000000 D2: 00000000 D3: 00000000 SSP: 00000000
D4: 00000000 D5: 00000000 D6: 00000000 D7: 00000000 Mem: 00000000:=00000000


Command:

===========================================================================
          —MIRAGE Debug Commands—–    Version 2.0    68000


LD - List Disassembled Code            FD - Full Disassembly to Disc
DM - Dump Memory in Hex & ASCII        DX - Display Exception Stack Frame
SM - Set Memory in Hex & ASCII          E - Execute MIRAGE Command
SR - Set Registers
SB - Set Breakpoint                    RD - Read Disc Sector
CB - Clear Breakpoint                  WR - write Disc Sector
DB - Display Breakpoints
ST - Set Command Line Tail             FM - Find Named or Known Module
GO - Execute at full speed             LM - List Known Modules
TR - Trace through code


 Q - Quit                              For more help,
 H - Help                              enter "H xx"
```

Bild 4: Das Menü des Debuggers

hergelaufene mal eben so den Rechner anwerfen kann, zum andern ist das File-Security-System eine feine Sache, wenn mehrere Rechner vernetzt sind. Man bleibt so sein eigener Herr über seinen Rechner. Gerade in letzter Zeit machen Computer-Viren wieder von sich reden. TOS macht es diesen kleinen Krabbeltierchen nicht besonders schwer, sich im System einzunisten. Unter MIRAGE haben es diese Parasiten nicht so einfach, verwendet man konsequent die erwähnten Möglichkeiten um seine Dateien abzuschotten. Nur wenige System-Benutzer haben überhaupt das Privileg physikalisch auf die Datenträger zugreifen zu dürfen, was aber für die heimliche Verbreitung der Viren Voraussetzung ist. Wenn darüberhinaus jeder Benutzer, der ein Programm oder eine Text-Datei fertiggestellt hat, diese - auch für sich selbst - gegen Überschreiben schützt, so wird jeder Änderungsversuch den ein Virus unternimmt, vom Betriebssystem zurückgewiesen.

Wem das aber alles zu umständlich ist, für den gibt es noch den sogenannten 'OPEN'-Modus, in dem keine Einschränkungen irgendwelcher Art bestehen.

Multitasking-/Multiuser

Die Multiuser-Fähigkeiten von MIRAGE sind schon mehrfach angesprochen worden. Voraussetzung hierfür ist natürlich, daß MIRAGE zwischen mehreren Tasks hin- und herschalten kann. Aber auch das allein reicht noch nicht aus, schließlich haben wir in unserem ST nur eine CPU, um die sich mehrere Programme schlagen. Die CPU-Zeit muß also möglichst sinnvoll aufgeteilt werden. Für dieses Problem gibt es mehrere Lösungsmöglichkeiten. Die beiden geläufigsten möchte ich kurz vorstellen:

Zum einen besteht die Möglichkeit, die CPU-Zeit in sogenannten Zeitscheiben zu vergeben. Bei dieser Methode wird ein Programm nicht an einem Stück abgearbeitet, sondern in einen bestimmten Zeitintervall. Ist diese Zeit um, so wird die Abarbeitung des aktuellen Programmes unterbrochen, und ein anderes aus der Warteschlange kommt an die Reihe. Im Leben eines Programms kann es aber auch Augenblicke geben, wo es freiwillig oder gezwungenermaßen eine Pause einlegt. Es wartet zum Beispiel auf einen Massenspeicher oder auf ein anderes Programm. In diesem Fall sorgt das Betriebssystem dafür, daß die Wartezustände des einen Programms in Rechenzeit für

```
MIRAGE MC68000 Execution Profiler Ver. 1.1.0

Frequency Profile for BENCH    ; Run of 14:45:30 on 14/06/88

Procedure INTMATH
  21:      10000 +****+****+****+****+****+****+****+****+****+*****+
Total Frequency is 10009 executions, 43.41% of Grand Total

Procedure REALMATH
  40:      10000 +****+****+****+****+****+****+****+****+****+*****+
Total Frequency is 10009 executions, 43.41% of Grand Total

Procedure TRIGLOG
  62:       1000 +*****+
Total Frequency is 1011 executions, 4.38% of Grand Total

Procedure STORE
  84:       1000 +*****+
Total Frequency is 1012 executions, 4.39% of Grand Total

Procedure TEXTSCREEN
 102:       1000 +*****+
Total Frequency is 1007 executions, 4.37% of Grand Total

Total of 6 Procedure(s)
Grand Total Frequency is 23057 executions
Plot Full Scale is 43.37% of Grand Total

MIRAGE MC68000 Execution Profiler Ver. 1.1.0

Time Profile for BENCH    ; Run of 00:45:30 on 03/03/03

Procedure INTMATH
  21:       5000 +*****+
Total Time is 5245 msec, 5.42% of Grand Total

Procedure REALMATH
  40:      10110 +****+****
Total Time is 10355 msec, 10.69% of Grand Total

Procedure TRIGLOG
  62:       5160 +*****+
Total Time is 5505 msec, 5.68% of Grand Total

Procedure STORE
  81:       1835 +**
  84:       8085 +****+**
  86:       1280 +*
  87:       8390 +****+***
Total Time is 19755 msec, 20.40% of Grand Total

Procedure TEXTSCREEN
 102:      55510 +****+****+****+****+****+****+****+****+****+*****+
Total Time is 55750 msec, 57.57% of Grand Total

Total of 6 Procedure(s)
Grand Total Time is 96845 msec
Plot Full Scale is 57.32% of Grand Total
```

Bild 5: Das Programm bench.pas wurde mit PROFILE untersucht

ein anderes umgewandelt werden.

Die Zeitscheiben-Methode hat den Nachteil, daß es bei zeitkritischen Programmteilen, die unbedingt an einem Stück ausgeführt werden müssen (Floppy-, Festplattenzugriffe, Abfragen einer Schnittstelle ...), dazu kommen kann, daß diese unterbrochen werden, was unter Umständen zum Zusammenbruch des Systems führen kann.

Damit es zu einem solchen Desaster nicht kommen kann, gibt man jedem Task eine Priorität. Dasjenige mit der höchsten Priorität erhält die CPU, solange keine Wartezustände hat oder freiwillig eine Pause einlegt. Ein 'Dauerrenner' - also ein Task ohne Wartezustände - läßt folglich für andere nichts mehr übrig. Für ein Multiuser-System ist dieser Ansatz also unbrauchbar.

MIRAGE bietet beide Möglichkeiten und vermischt sie teilweise. Es kennt den 'normalen' Task, der eine Zeitscheibe zugeteilt bekommt und eine Priorität hat. Außerdem bietet MIRAGE noch Echtzeittasks in zwei Varianten an. Darunter versteht MIRAGE ein Task, das nur von einem höher priorisierten Echtzeittask, von Wartezuständen oder durch freiwilligen Verzicht unterbrochen werden kann, auf das also die Zeitscheiben keinen Einfluß haben.

Nun gibt sich MIRAGE mit einfachen Tasks aber nicht zufrieden. Vor das Task haben die User nämlich noch die Jobs gesetzt. Jobs sind unabhängige Funktionseinheiten, die in den Weiten der MIRAGE-Welt ihr Dasein fristen. Der Unterschied zu den Tasks ist nur der, daß ein Task immer einen Vater hat, der es ins Leben gerufen hat, ein Job dagegen sozusagen à la Klapperstorch vom Himmel gefallen ist. Ein Job kann bis zu 128 Tasks im Vater-Sohn-Prozeß erzeugen, ein Task kann das seinerseits nun auch, und schon haben wir wieder die schönsten Baumstrukturen, wie sie ja in der ST-Computer schon des öfteren behandelt wurden.

Jeder Job verwaltet seinen eigenen Speicherbereich und sein eigenes - auch virtuelles - Terminal. Ein Task wird mit oder ohne virtuellen Bildschirm erzeugt. Mit dem FAKIR-Programm kann man bei der ST-Version mittels Tastenkombination oder mit der rechten Maustaste zwischen den virtuellen Bildschirmen zyklisch umschalten. Dieses FAKIR-Programm schränkt allerdings den ATARI-ST-Anwender auf 7 Jobs ein. Prinzipiell ist die Zahl der Jobs nur durch den Speicherplatz beschränkt.

Das ganze Konzept von MIRAGE ist auf Multiuserbetrieb ausgelegt, weniger auf Echtzeitbetrieb. Dafür gibt es spezielle Betriebssysteme und Sprachen, die für diesen Zweck entwickelt worden sind. Ich denke da besonders an die Kombination RTOS/PEARL. Hier wurde für die Echtzeitsprache PEARL ein passendes Betriebssystem geschrieben. Mit MIRAGE an sich ließen sich Echtzeitprobleme durchaus lösen - allerdings auf Assemblerebene und nicht in einer Hochsprache. Es wäre vermutlich möglich, einen PEARL-Compiler für MIRAGE zu entwickeln oder PASCAL und FORTRAN um Echtzeiteigenschaften zu erweitern, doch bis jetzt ist von alledem nichts in Sicht.

Interrupts

Beim 68000 kann es in Multitasking-/Multiuser-Systemen Probleme geben. Alle Interrupts müssen beendet sein, bevor ein Task-Wechsel stattfindet. Datei werden der User- und Supervisor-Stackpointer gerettet und jeweils ein neuer eingerichtet. Da die Interruptroutinen aber ihrerseits den Supervisor-Stack benutzen und ebenfalls Daten auf diesem ablegen, werden diese bei einem Task-Wechsel mit ausgetauscht. Die Interruptroutine hängt dann quasi in der Luft, allerdings ohne Netz und doppelten Boden - ein System-Crash ist die Folge. Beim 68020 hat man dieses Problem durch einen eigenen Interrupt-Stack gelöst. Für den 68000 haben sich die Entwickler von MIRAGE eine Softwarelösung einfallen lassen: Alle Interruptanforderungen werden an einen Interrupt-Manager übergeben. Dieser unterdrückt einen Task-Wechsel, solange noch ein Interrupt bei der Arbeit ist.

Für den normalen Anwender hat das weiter keine Bedeutung, nur der Systemprogrammierer muß sich halt an die System-Aufrufe halten und darf die Interrupt-Vektoren nicht direkt setzen.

Grafik?

Für viele eine besonders leckere Mischung ist die Kombination von Grafik und Multitasking. Wäre es nicht herrlich, vor 1st Word zu sitzen, einen Text zu schreiben und gleichzeitig im Hintergrund ein Apfelmännchen zu berechnen? MIRAGE kann das! Natürlich nicht mit 1st Word, wohl aber mit dem integrierten Text- und Kalkulationspaket OxSys-Suite. Moment, Moment - ich denke, MIRAGE kennt nur reine Textterminals? Richtig, aber es lassen sich die LINE-A-Routinen des TOS aufrufen!

Sogar mit virtuellen Bildschirmen, womit gewährleistet ist, daß der Grafikaufbau auch wirklich im Hintergrund stattfindet. Für diesen Grafikzauber sind allerdings 164 deutsche Mark zu berappen. Dafür erhält man dann aber eine vollständige LINE-A-Bibliothek, die sowohl unter der TOS-APL-Version wie auch unter MIRAGE nutzbar ist und umfangreiche Makros enthält. So ist es möglich, eine Liste mit den Koordinaten aller für die Grafik nötigen Vektoren zu erstellen und diese Liste dann mit einem einzigen Aufruf einer Routine aus dieser Bibliothek abzuarbeiten. Das kommt APL besonders entgegen, denn in dieser Sprache werden Schleifenkonstrukte vergleichsweise langsam abgearbeitet, dagegen sind Operationen auf Strukturen sehr schnell, da die sequentielle Abarbeitung in Assembler kontrolliert wird. Dieses Tool ist von gdat in Bielefeld entwickelt worden und wird mit einem deutschen Handbuch geliefert,das sich angenehm vom Durchschnitt abhebt. Unverkrampft wird der Leser mit den Möglichkeiten seiner neuen Erungenschaft vertraut gemacht, immer wieder aufgelockert durch gelungene Gags - es macht einfach Spaß, diese Anleitung zu lesen.

Drum und Dran

Fassen wir kurz zusammen: Da haben wir ein Multitasking-/Multiuser-Betriebssystem, das ordentlich aufgebaut ist, seit mehreren Jahren auf Workstations läuft und nun auch relativ preisgünstig für den ATARI-ST verfügbar ist. Doch was gibt es außer dem splitterfasernackten Betriebssystem mit seinen - zugegeben umfangreichen - Utilities?

Antwort: Leider nicht viel. Es gibt nur ein einziges Anwendungspaket, ein integriertes Software-Paket, das aus einer Textverarbeitung, einer Tabellenkalkulation, einem speziellem Druckprogramm und einem Programm zur automatischen Generierung eines Inhaltsverzeichnisses besteht. Neuerdings ist für dieses Paket noch ein objektorientiertes Zeichenprogramm - ähnlich EASYDRAW - erhältlich. Dieses Paket macht einen professionellen Eindruck, doch lag es zum Test leider nicht vor. Erwähnenswert ist, daß Wordstar-Files von allen Programmteilen akzeptiert werden.

Wer dagegen Software unter MIRAGE entwickeln möchte, kann sich die Sprache aussuchen:

ALPS (Assembler-Language Programming System)

Als Assembler-Freak kommt man hier voll auf seine Kosten. Ein Makro-Assembler, der wirklich (fast) keine Wünsche mehr offen läßt (sehr schnell, 68000er, 68020er und 68881 Code). Ein komfortabler Debugger (Bild 4) gehört ebenso zu diesem Paket wie ein Programm namens PROFILE. Mit diesem Tool kann man die Aktivitäten von Hochsprachen-Programmen (nur von den Swifte-Sprachen PASCAL, FORTRAN und BASIC) genau verfolgen. So wird die Anzahl der Aufrufe und die verbrauchte Zeit jeder Prozedur aufgelistet (Bild 5). Diese Information ist nützlich, wenn man Fehler suchen oder sein Programm optimieren muß. Ein Linker rundet dieses Paket ab. Bei der Profiversion kommen noch verschiedene Bibliotheken und Beispiele dazu.

Verschiedene Softwarehäuser in England nutzen das ALPS-Paket schon als Entwicklungssystem für TOS-Programme. So wurde zum Beispiel ARMAGEDDON-MAN von Electric Pencil unter MIRAGE geschrieben.

Swifte-BASIC

Umfangreicher BASIC-Compiler mit einigen Extras wie Subroutinen-Aufruf mit Parametern oder einer Redimensionierungsmöglichkeit von Feldern. Ansonsten halt doch eben BASIC.

Swifte-FORTRAN-77

Vollständiger ANSI X3.9-1978-Compiler mit NCC ERROR-FREE-Zertifikat, was immer das auch heißen mag. Schön ist, daß der Compiler auch 68020- und 68881-Inlinecode erzeugen kann, was für PAK-68-Besitzer interessant sein dürfte. Mit verschiedenen Optionen kann man den Compiler beeinflussen, zum Beispiel läßt sich der ANSI-66-Standard einstellen oder die Bereichsüberprüfung von Feldern ausschalten.

Ein besonderer Leckerbissen sind die verfügbaren NAG-Libraries, um die sich angeblich die FORTRAN-Leute schlagen. Ich muß gestehen, daß ich zu FORTRAN kein enges Verhältnis habe, mehr FORT als RAN, aber viele Programme sind in dieser Sprache geschrieben, und sie ist in der Wirtschaft und an Universitäten weit verbreitet. FORTRAN-Anwender haben hier ein zuverlässiges System in der Hand, um zum Beispiel Programme vom Großrechner zu übernehmen und damit diesen zu entlasten.

Bezugsnachweis

MIRAGE wurde und wird von SAHARA-SOFTWARE-LIMITED in London entwickelt. In Deutschland wird MIRAGE von gdat in Bielefeld vertrieben. Die deutsche Firma bietet einen umfangreichen Update-Service; dank der guten Beziehung zu SAHARA-SOFT werden spezielle Anpassungsprobleme sofort weitergeleitet und dann auch wirklich behoben. In meinem Rechner habe ich seit kurzem eine PAK-68 (68020-Karte mit 68881) eingebaut, die leider unter MIRAGE nicht funktioniert. Ein Anruf bei gdat ergab, daß definitiv mit dem nächsten Update die PAK unterstützt wird. Ich möchte an dieser Stelle die PAK-Anwender dazu auffordern, bei den Softwareherstellern Druck zu machen, denn so schwierig kann es nicht sein, die rechenintensiven Programme wie zum Beispiel CAD- oder Grafikprogramme auf 68020 und Coprozessor umzustellen. Doch nun zu den lästigen Kleinigkeiten, die der Erwerb von MIRAGE mit sich bringt - ein größeres Loch im Geldbeutel nämlich.

MIRAGE-ST	Betriebssystem	348.-
MicroGrid	MIDI- und RS232-Netzwerksoftware	98.-
ALPS	Compilerversion	100.-
	Profiversion	185.-
	Update von Compiler- auf Profiversion	85.-
Swifte-Basic	Basic-Compiler	248.-
Swifte-Fortran-77	Error-Free-Zertifikat	398.- *)
Swifte-Pascal	ISO/BSI-Standard	248.- *)
Lattice C	mit Erweiterungen für MIRAGE	386.- *)
APL	unterstützt Multitasking	498.-
Swifte-Lisp	Interpreter	348.-
OxSys Suite	Integriertes Text- Kalkulationspaket	798.-
Forth		???

Die mit *) gekennzeichneten Sprachen Fortran-77, Pascal und Lattice C benötigen noch die Compilerversion des ALPS, da die Compiler Assemblercode erzeugen, der mit dem Makro-Assembler noch übersetzt werden muß. Es kommen für diese Sprachen also noch einmal zusätzlich 100 DM dazu. Natürlich nur einmal, wenn man in zwei oder allen drei Sprachen programmieren möchte.

Swifte-PASCAL

Zuverlässiger Compiler mit vielen Erweiterungen - zum Beispiel dynamische Strings. Es besteht die Möglichkeit zur Laufzeit Sub-Tasks zu generieren und FORTRAN-Programme einzubinden. Verschiedene Compiler-Optionen ermöglichen es, wie bei FORTRAN die verschiedenen Überprüfungen zur Laufzeit auszuschalten. Ebenso kann man auf reinen ISO-Standard schalten, bei dem sämtliche Erweiterungen wegfallen. Im Handbuch wird unter 'Program Optimisation' übrigens darauf hingewiesen, daß man rechenaufwendige Programmteile doch möglichst in FORTRAN schreiben soll, denn PASCAL ist mehr eine Sprache für allgemeine Anwendungen und kann bei Fließkomma-Berechnungen niemals so schnell sein wie FORTRAN - angeblich. Das möchte ich lieber nicht kommentieren, ich überlasse das den PASCAL-Anwendern...

C (Lattice)

Ist nach Aussagen von gdat nicht optimal an MIRAGE angepaßt (Originalton: 'Kröte'), aber immerhin gibt es C.

APL/68000

Spezielle Version für MIRAGE, d.h. für Multitasking-Betrieb ausgelegt. Für den Multiuser-Betrieb mit mehreren Terminals nur eingeschränkt zu verwenden - aber wer hat schon mehrere Terminals an seinem ST? Die TOS-Version von APL ist allerdings komfortabler.

Swifte-Lisp

Ich hatte es zwar zum Test, doch muß ich zugeben, daß ich von dieser Sprache überhaupt keine Ahnung habe. Deshalb kann ich nur sagen: Es gibt Lisp unter MIRAGE.

See you later, alligator

Viel gäbe es noch zu sagen, manches ist nur oberflächlich gestreift worden, doch irdendwo muß ich einen Schlußstrich ziehen, sei es, weil MIRAGE einfach zu komplex ist, oder weil ich auf manchen Gebieten nicht genug Ahnung habe. Bei meiner Arbeit mit MIRAGE habe ich festgestellt, daß es keineswegs leicht ist, ein Betriebsystem zu bewerten. Während meines Test gab es keine Abstürze, nur einmal muß ich das System etwas arg gebeutelt haben, als ich in FORTRAN einen Benchmark schreiben wollte und ein falsches Format für die Ausgabe von ganzen Zahlen verwendet habe. Seine Herkunft aus Großbrittanien kann MIRAGE nicht verleugnen: So leistungsfähig es auf der einen Seite ist - so hat es auf der anderen Seite eine ganze Menge Eigenheiten, an die es sich zu gewöhnen gilt. Ich denke da zum Beispiel an das Diskettenhandling. Sein Geld ist es auf jeden Fall wert, obwohl es nicht ganz billig ist, denn für MIRAGE, den Macro-Assembler und eine Hochsprache muß man schon 800-900 DM anlegen (siehe Kasten). Rechnet man allerdings den Service der deutschen Vertriebsfirma hinzu, die viele Hilfestellungen und regelmäßige Updates gibt, so ist diese Summe sicher nicht zuviel.

Für wen?

Eine schwierige Frage. Ich könnte es mir leicht machen und sagen: Für den, der's brauchen kann. Doch das wäre etwas zu einfach. MIRAGE ist sicher nichts für denjenigen, der mit BASIC wunderbar zurechtkommt und nur mal eben seine BASIC-Programme im Multitasking-Betrieb laufen lassen will. Solche ST-Benutzer sind mit TOS besser bedient. Im Laufe des Tests habe ich an verschiedenen Stellen Personengruppen genannt, für die MIRAGE interessant ist. Ich möchte sie noch einmal kurz nennen:

Programmentwickler, die einen Profi-Makro-Assembler suchen und/oder im Team zusammen an einem Rechnernetz arbeiten wollen.

Studenten, Professoren und andere an der Uni Beschäftigte, die unab-hängig vom Großrechner ihre FORTRAN-Programme entwickeln wollen.

Universitäten und andere Schulungseinrichtungen, die einen preiswerten Rechner wie den ATARI-ST unkompliziert und mit wenig Aufwand zu einem Netz zusammenschließen möchten.

C.D.Ziegler

Bezugsadresse:

gdat
Stapelbrede 39
4800 Bielefeld 1
Tel.: 0521/875888

ENDE

SUPERSCORE

Ein Notenlayoutprogramm

Auf der Frankfurter Musikmesse gab es an einem kleinen, in einer Ecke versteckten Stand ein interessantes Produkt zu sehen, das endlich auch Erleichterung für die handschriftgeschädigten Komponisten und Arrangeure bringen soll.

Der kleine Unterschied zu den meisten anderen für den ST verfügbaren Programmen dieser Gattung ist der, der zwischen einer Textverarbeitung wie 1st Word und einem Desktop Publishing-Programm besteht: Ersteres erlaubt es, einfach und schnell große Mengen von Text zu erfassen und in brauchbar präsentable Form zu bringen. Ein DTP-Programm dagegen soll dem Layout, der grafisch möglichst vollkommenen Fassung der meist anderswo erfaßten Texte dienen. SuperScore ist ein Programm, das es ermöglicht, auch musikalisch komplexe Noten zu gestalten und zu drucken. Für die Eingabe von Noten wurde ein 32-spuriger Sequencer gleich in das Programm integriert. Man darf gespannt sein, ob das Gespann sich bewährt.



SuperScore wird in einem Plastikschuber geliefert, blaumetallic, in üblicher Programmschubergröße. Darinnen findet man Handbuch, Programmdiskette und einen Hardware-Key, der als Kopierschutz fungiert. Vor dem Einschalten des Rechners muß dieser Key in den Romport gesteckt werden - eine Kopierschutzmethode, die sich inzwischen bei Musikprogrammen durchgesetzt hat. Man benötigt mindestens 1 Megabyte RAM und, man höre und staune, für ein amerikanisches Programm höchst ungewöhnlich, einen Schwarzweiß-Monitor. Farbmonitor-Besitzer dürfen sich also mit Fug und Recht diskriminiert fühlen. Zusätzlich zum Rechner brauchen Sie nur noch ein Midi-Keyboard als Eingabe- (und Wiedergabe-) Gerät.

Das Handbuch ist natürlich englisch geschrieben und nicht besonders übersichtlich. Es leidet auch etwas unter der Tatsache, daß sich der eigentliche Text auf die Version 1.0 bezieht, während die ausgelieferte Version bereits die Nummer 1.2 trägt. Die zusätzlichen bzw. geänderten Features sind auf losen Anhangblättern beschrieben, jedoch leider nicht vollständig. Der deutsche Vertrieb hat aber zugesichert, daß sowohl ein deutsches Handbuch wie eine deutsche Übersetzung des Programmes selbst in Arbeit sind.

Apropos Übersetzung, die Programmierer von SuperScore haben eine gute Idee in Sachen Übersetzbarkeit gehabt:

Sämtliche Meldungen und Dialoge, die das Programm so von sich gibt, sind in einem ASCII-File enthalten, das vom Benutzer nach Wunsch geändert werden kann. Einer französischen oder dänischen Fassung steht also nichts im Wege: einfach das Resource-File und das File mit den Meldungen in die gewünschte Sprache übersetzen. Kein neues Compilieren ist notwendig; damit sollten schlechte Übersetzungen endlich der Vergangenheit angehören. Nachdem man den Key in den Romport gesteckt hat, sind keine weiteren Installationen notwendig; natürlich sollte man, wie immer, eine Sicherheitskopie anfertigen. Problemlos ist es auch, das Programm samt allen Hilfsdateien auf eine Harddisk zu kopieren; da SuperScore jedoch nach Wunsch auch mit GDOS-Fonts (für Titel und Texte) arbeiten kann, muß eine Zeile im ASSIGN.SYS-File geändert werden, das ja den Zugriffspfad und die Namen der Fonts, die von GDOS geladen werden

sollen, enthält.

Die Bedienung von SuperScore ist mausorientiert; jedoch kann man die Tastatur nicht in die Ecke stellen - für eine Menge Tätigkeiten werden Tastatur-Eingaben verlangt. Die meisten Menü-Kommandos stehen für gedächtnisstarke Benutzer auch als Tastenfunktionen zur Verfügung.

Wie bereits in der Einleitung angedeutet, besteht SuperScore aus zwei Teilen: einem 32-spurigen Midi-Sequencer, der der Erfassung der Noten dient und dem eigentlichen Notensatz-Teil, der sich dem Layout der Musik verschrieben hat.

Der Sequencer

Das Konzept dieses Programmteils ist einfach und dem Zweck angemessen. Eine Aufnahme kann aus insgesamt 24 Sequenzen, jede bestehend aus 32 Spuren, zusammengesetzt werden. Sequenzen können beliebig aneinandergebunden werden, so daß man eine längere Aufnahme bequem aus einzelnen Teilstücken zusammensetzen kann. Der Inhalt jeweils einer Sequenz kann dann in den Notensatz-Teil übernommen werden.

Die Länge jeder Sequenz richtet sich nach der Länge der ersten Aufnahme auf einer Spur. Wenn Sie also in einer leeren Sequenz auf Spur 4 genau fünf Takte aufnehmen, hat diese Sequenz eine Länge von fünf Takten.

Apropos Aufnahme: Sie funktioniert wie bei einem Tonbandgerät. Sie wählen Sequenz und Spur aus, auf die Sie aufnehmen möchten, und drücken die Record-Taste. Das 'Band' - sprich der Sequencer - läuft los und stoppt erst, wenn Sie die 'Stopp'-Taste drücken. Dabei können Sie ein Metronom mit einstellbarem Vorzähler aus dem Lautsprecher des Atari hören. Für die erste Spur ist die Aufnahme noch komfortabler: Das Metronom läuft vor, die Aufnahme beginnt genau in dem Moment, in dem Sie die erste Taste auf Ihrem Midi-Keyboard drücken.

Zur Korrektur eventueller Spielfehler gibt es ein manuelles und ein automatisches Punch-In/Out. Sie können also von einer beliebigen Stelle einer Aufnahme an noch einmal beginnen und damit Fehler verbessern.

Schließlich ist es mit dem sogenannten Step-Modus möglich, Musik nicht einzuspielen, sondern Note für Note einzugeben. Man kann die Midi-Clock dabei automatisch oder manuell fortschreiten lassen, so daß auch und gerade sehr komplizierte Rhythmen schnell in den Rechner zu bringen sind. Sehr nützlich ist außerdem, daß man die Step-Eingabe auch in Verbindung mit den Punch-In/Out-Verfahren für Korrekturen benutzen kann.

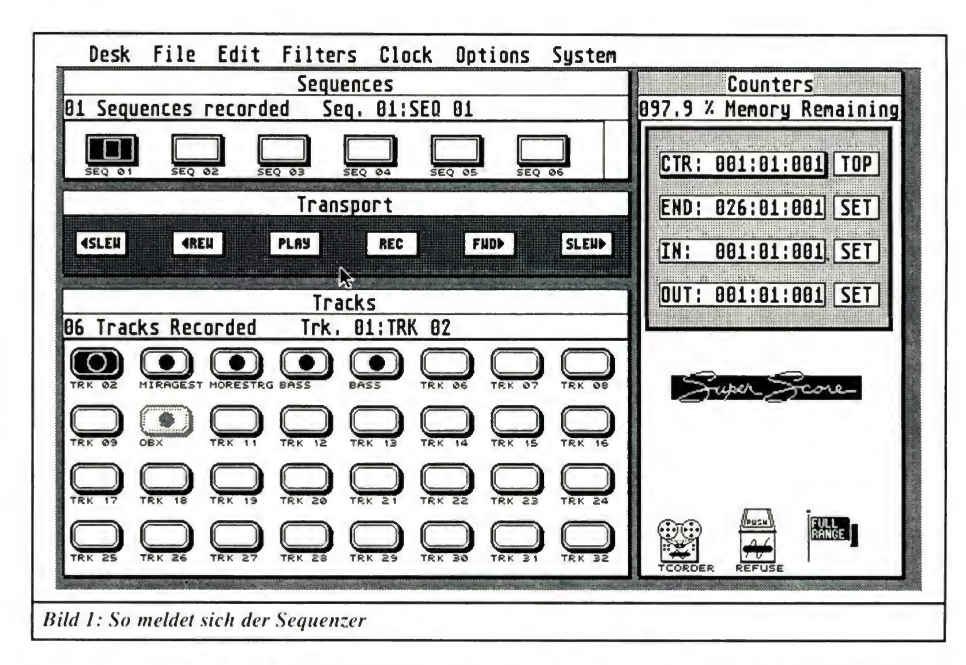


Bild 1: So meldet sich der Sequenzer

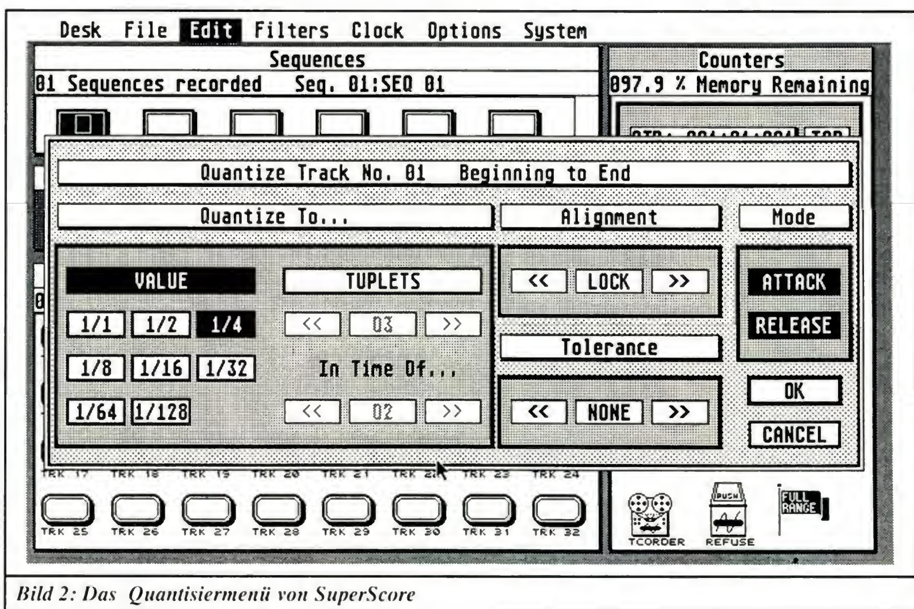


Bild 2: Das Quantisiermenü von SuperScore

Wie sich das für einen Sequencer gehört, laasen sich Aufnahmen natürlich auch wieder abspielen. Dabei kann jeder Spur ein eigener Midi-Kanal zugeordnet werden, mit einem kleinen Hardware-Zusatz kann man den Atari sogar mit drei zusätzlichen Midi-Ausgängen versehen, so daß maximal 64 Midi-Kanäle verwendet werden können. Jeder Spur ordnet man einen Ausgang und ein Midi-Kanal zu. Sequenzen können im Loop-Modus abgehört werden, so daß die Aufnahme immer wiederholt wird. Natürlich existiert auch das Vor- und Zurückspulen, wie bei einem Kassettenrekorder. Das Tempo ist aber, im Gegensatz zu einem richtigen Rekorder, frei wählbar, wie bei Midi-Rekordern üblich. Eine Solo-Funktion ist vorhanden, um einen einzelnen Track alleine abzuhören. Jede Spur läßt sich per Hand an- und ausschalten.

Ganze Sequenzen und auch einzelne Spuren sind nach der Aufnahme bearbeitbar: Wie gesagt, können Sequenzen aneinandergehängt werden. Darüber hinaus kann man sie kopieren, ihren Inhalt um einige Takte oder auch nur einige Schläge verschieben, kürzen oder auch transponieren. Die Transponierfunktion ist recht intelligent: Es lassen sich einzelne Midi-Kanäle von der Transposition ausnehmen, was besonders für Schlagzeugspuren wichtig ist.

Einzelne Spuren können zuerst einmal quantisiert werden. Unter Quantisierung versteht man eine Korrekturfunktion des Computers, bei der der Rechner kleine Spielungenauigkeiten auszugleichen ver-

sucht. Die Quantisierung ist recht vielseitig: Sowohl der Quantisierungs-Notenwert wie auch eine gewisse Toleranz können gesetzt werden, ebenso hat man die Wahl, ob Notenanfang oder -ende oder auch beide korrigiert werden sollen. Auch die Midi-Velocity-Werte (Anschlagsdynamik) können verändert werden.

Wie ganze Sequenzen lassen sich auch einzelne Spuren verschieben oder transponieren. Spuren können gemischt und (nach Midi-Kanälen) wieder entmischt werden. Auch eine tonhöhenabhängige Aufteilung auf mehrere Spuren ist möglich, um z.B. eine Klavieraufnahme auf zwei Spuren zu verteilen (linke und rechte Hand). Schließlich kann man noch bestimmte Midi-Daten löschen, z.B. Controller- oder Pitch Bending-Daten.

Natürlich gibt es auch eine Filterfunktion beim Einspielen, so daß unliebsame Daten gar nicht erst aufs elektronische 'Band' kommen. Eine Menge nützlicher Funktionen sind noch vorhanden, ein kleines Midi-View-Utility erlaubt es zum Beispiel, ankommenden Midi-Daten zuzuschauen, was sehr nützlich ist, um Fehler in einem Midi-Setup zu finden.

Selbstverständlich kann der Sequencer auch extern synchronisiert werden, ebenso kann er selbst ein Midi-Clock-Signal senden. Auch eine Software-Midi-Thru-Funktion ist vorhanden.

Der Sequencer arbeitet sauber und problemlos, seine Bedienung ist, mit wenigen kleinen Ausnahmen, sehr einfach und übersichtlich. Seine Bedieneroberfläche ist stark fensterorientiert, was die vom MacIntosh bekannte Unannehmlichkeit mit sich bringt, daß natürlich nie das Fenster aktiv ist, in dem man gerade etwas anklicken will. Viele Operationen gehen wie auf dem Desktop schnell und einfach, so zum Beispiel das Kopieren von Spuren: einfach mit der Maus nehmen und auf die gewünschte Zielspur ziehen - wie das Dateienkopieren des Desktops.

Wenn man bedenkt, daß der primäre Zweck des Sequencers die Eingabe von Noten für das Layout-Programm ist, so hat er eine hervorragende Note verdient. Natürlich kann er sich nicht mit State-of-the-Art-Sequencern messen, aber das ist auch nicht seine Aufgabe. Man kann gut und schnell damit aufnehmen, auch Arrangements sind durch das sequenzorientierte Arbeiten schnell gemacht. Man nimmt die Einzelteile in getrennten Sequenzen auf und hängt sie dann einfach in der gewünschten Reihenfolge hintereinander. Für viele Anwendungen gerade im Heimstudio dürfte der Sequencer völlig ausreichen.

Wenn Sie eine Aufnahme fertiggestellt haben, wechseln Sie auf die ScoreWriter-Seite. Dort beginnt der Computer damit, Ihre Aufnahme in ein Seitenlayout umzurechen.

Dazu können Sie einige Voreinstellungen machen. Wie bei jedem Layout-Programm muß der Computer zuerst einmal wissen, welches Format - in diesem Fall, speziell welches System-Format - Sie für Ihre Noten haben wollen. Sie können also wählen. Hier legt SuperScore eine ungeahnte Vielseitigkeit an den Tag, die einzige Beschränkung ist das Papierformat: 8*11 Zoll; also etwas kleiner als DIN A4 ist festgelegt.

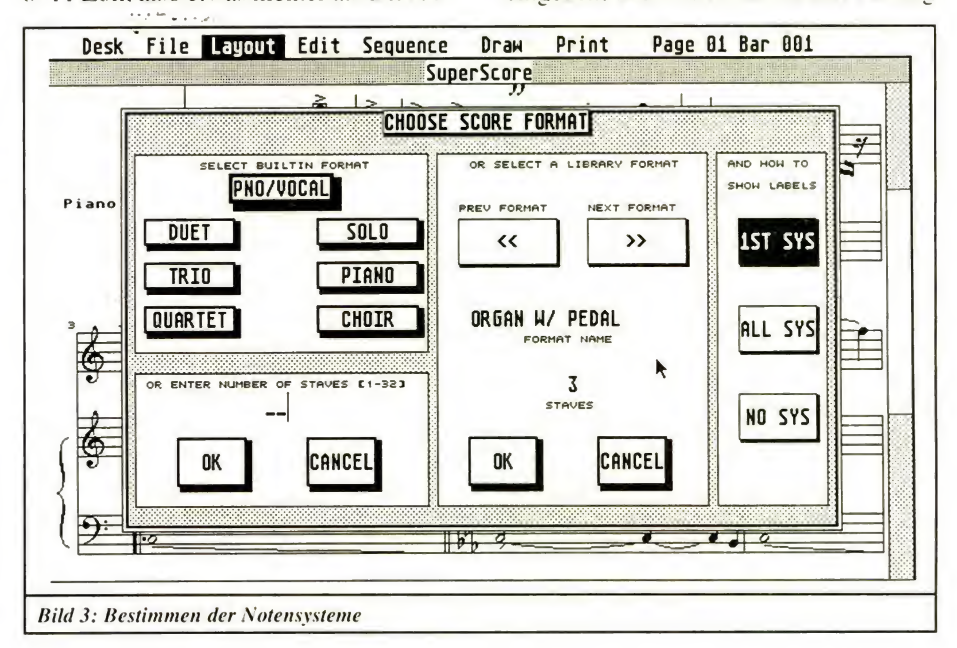


Bild 3: Bestimmen der Notensysteme

Sieben eingebaute Formate stehen zur Verfügung, die wohl die meisten Standard-Anforderungen erfüllen dürften (z.B. Klavier, Klavier und Solostimme, Trio, Chor usw.). Wenn Ihnen keines dieser Formate zusagt, können Sie auch einfach eine Anzahl von Systemen festlegen, maximal 32 Stimmen sind möglich. Als besonderen Gag gibt es aber eine Bibliothek mit Systemformaten, die Sie selbst beliebig verändern und erweitern können, denn die Bibliothek ist, wie auch die Programmeldungen, als ASCII-File auf der Diskette gespeichert. Wenn Sie für Ihren Chor oder Ihre Funk-Band ein bestimmtes Notenformat brauchen, tragen Sie es einfach in die Bibliothek ein, und es steht Ihnen immer auf Abruf zur Verfügung. In dieser Bibliothek sind auch gleich (natürlich veränderbar) die Zuordnungen von Notensystemen zu Sequenzerspuren sowie die Bezeichnungen der Spuren samt Klammern und Akkoladen und schließlich auch die Art des Notenschlüssels verzeichnet, kurz - das ganze Systemlayout.

Spurenzuordnung, Notenschlüssel und Systembezeichnungen sind jederzeit veränderbar, mit den Akkoladen und Klammern ist das nicht ganz so einfach. Es gibt übrigens sechs verschiedene Schlüssel: Sopran, Alt, Tenor und Bass, sowie oktavierten Sopranschlüssel und die Möglichkeit, ganz auf einen Schlüssel zu verzichten, z.B. für Schlagzeug.

Weitere Voreinstellungen betreffen die Größe Ihrer Noten. Es gibt klein, mittel und groß, mehr nicht. Natürlich können Sie auch Tonart und Taktbezeichnung eingeben, aber nicht nur für den Anfang des Stückes, sondern für jeden beliebigen Takt. Die Tonart läßt sich für jedes System einzeln einstellen, die Taktart nur für alle gemeinsam. Die Taktnumerierung kann ebenfalls frei angegeben werden.

Schließlich sollten noch Titel und Kopf- und Fußzeilen eingegeben werden. Dabei können Schrifttyp und -größe aus den mit GDOS eingeladenen Fonts ausgewählt werden (wie bei DEGAS ELITE). Die Zeilen für Titel usw. sind ausreichend, für jeden Teil (Titel, Untertitel, Orgelregister, Komponist, Texter, Aufführungshinweis) ist ein eigener Font bzw. Schrifttyp wählbar. Kopf- und Fußzeilen können übrigens entweder für alle Seiten gemeinsam (mit unterschiedlicher Numerierung) oder auch für jede Seite unabhängig eingegeben werden. Leider gibt es bei der Titelformatierung manchmal Probleme, vor allem, wenn sie den Titel erst nachträglich angeben.

Übrigens haben Sie auch die Möglichkeit, auf der Layout-Seite Ihre Noten jederzeit abzuhören. Die Abspiel-Kommandos des Sequenzers sind auch auf der Layout-Seite verfügbar.

Die bisherigen Voreinstellungen sind ziemlich elementar und können auch schon eingestellt werden, bevor überhaupt irgendwelche Noten aufgenommen werden. Das ist auch sinnvoll, weil viele der spezielleren Edierfunktionen Änderungen an diesen Grunddaten nicht überleben. Man sollte wirklich immer erst das Grundlayout endgültig fertigstellen, bevor man zu den verschiedenen Ebenen von Details, die mit SuperScore verändert werden können, übergeht. Jede nachträgliche Änderung auf einer höheren Ebene kann die ganze Arbeit auf den niedrigeren Ebenen zunichtemachen. Klar, wenn Sie z.B. in einem DTP-Programm das Spaltenformat wechseln, gehen alle Änderungen, die Sie an der Silbentrennung oder am Kerning (dem Buchstabenabstand innerhalb eines Wortes) vorgenommen haben, verloren, bzw. sie verlieren ihren Sinn.

Nach diesen Grundsätzlichkeiten sollte der nächste Schritt die ungefähre Verteilung der Systeme auf der Seite sein. Dazu gibt es drei Funktionen:

'Score Quantize', für jedes System einzeln wirksam, bestimmt, wie groß der kleinste noch dargestellte Notenwert sein soll. Wenn Sie hier z.B. ein Achtel einstellen, werden alle Noten, die bei der Aufnahme kürzer als eine Achtelnote gespielt wurden, nicht dargestellt bzw. zu Achteln verlängert. Diese Funktion ist unter anderem notwendig, um Unterschiede zwischen Aufführungspraxis und Notation zu beheben (Woher soll der Computer auch wissen, daß Sie die 16tel, die Sie gespielt haben, als Staccato-Achtel notiert haben wollen ?).

Mit 'Horizontal Spacing' können Sie die einzelnen Noten enger zusammenrücken lassen oder aber auch mehr Platz für Ihre Entfaltung schaffen. Mehr Platz ist meistens dann notwendig, wenn Text unterlegt werden soll. Weniger Platz wird man meistens dann verwenden wollen, wenn möglichst viele Takte in ein System passen sollen.

Mit 'Vertical Spacing' können Sie den Abstand jedes Systems von den anderen völlig frei einstellen. Allerdings benutzt SuperScore diese Einstellungen nicht als absolute Werte, sondern eher relativ; in gewissen Grenzen behält sich das Programm die endgültige, präzise Formatierung der Systeme selbst vor. Wie in den meisten gedruckten Noten auch ist der Abstand zwischen zwei Systemen normalerweise auch nicht völlig gleich, sondern von dem enthaltenen Tonmaterial abhängig; Wenn z.B. nur an einer Stelle ein Ton mit vielen Hilfslinien vorkommt, der einen großen Systemabstand erfordert, wird SuperScore an dieser Stelle automatisch einen ausreichend großen Systemabstand wählen, so daß es nicht zu Überlappungen kommt. An anderen Stellen verschwendet SuperScore den Platz aber nicht, sondern reduziert den Systemabstand wieder.

Wenn Sie unbedingt wollen, können Sie natürlich mit der 'Vertical Spacing'-Option auch Überlappungen zwischen Systemen erzeugen, Ihrer Fantasie sind da keine Grenzen gesetzt. Ziemlich mühsam wird die Art der Formatierung bei längeren Werken, da es keine Option gibt, mit der sich der Vertikal-Abstand für die ganze Partitur relativ vergrößern oder verkleinern ließe.

Schließlich kann man das Layout ganzer Seiten beeinflussen, indem man ganze Systeme von der Seite entfernt oder hinzufügt, falls der übrige Platz das erlaubt. SuperScore berechnet das neue Layout nicht nach jedem dieser Formatierkommandos. Man kann zwar die neuen Positionen erkennen, die alten Daten werden aber nicht immer gelöscht, was manchmal etwas unübersichtlich ist. Deshalb gibt es ein Kommando, mit dem dann zum krönenden Abschluß das Layout komplett neu berechnet und gezeichnet werdenkann.

Jetzt ist man glücklich an dem Punkt angelangt, an dem SuperScore seine Vorteile gegenüber anderen Notendruckprogrammen ausspielen kann: Das Aussehen jeder einzelnen Note kann geändert werden, und zwar nicht nur in Sachen Tonhöhe und -länge, auch die Art der Balken und die Richtung der Notenhälse (und vieles mehr) ist veränderbar. Weshalb das notwendig ist, mag ein Beispiel erläutern:

Bei Vokal-Noten ist es üblich, jede Textsilbe mit einem eigenen Fähnchen zu zeichnen, auch wenn die normale Regel in der entsprechenden Taktart eigentlich Balken statt Fähnchen verlangte. Ein Programm, das Balken nur automatisch setzt, wie die meisten Notendruckprogramme, ist an diesem Punkt überfordert. Das gleiche Problem tritt bei Taktarten auf, für die es keine oder keine eindeutigen Regeln für die Balkensetzung mehr gibt, wie z.B. schon ein einfacher 5/4tel Takt. Oder: Bei polyphoner Musik kann der Computer nicht entscheiden, welche Noten zu welcher Stimme gehören. In diesen Fällen muß es möglich sein, die Richtung von Notenhälsen per Hand zu bestimmen. Oder: Der Computer kann an eingespielten Noten nicht erkennen, ob es eventuell Schlagzeug- oder Flageolett-Noten sein sollen. Auch manche N-Tolen sind für den Computer unter Umständen automatisch nicht zu erkennen. Sie sehen, es gibt so einiges, was man manuell an einem Notenblatt zu layouten hat.

Kurz gesagt, erlaubt SuperScore, jede Note zu löschen, beliebig zu verschieben, in Tonhöhe, Länge, Bebalkung, Notenhälsen und Bindebögen beliebig zu manipulieren. Selbst die Länge der Notenhälse kann frei eingestellt und Vorzeichen können unabhängig von der Note, zu der Sie

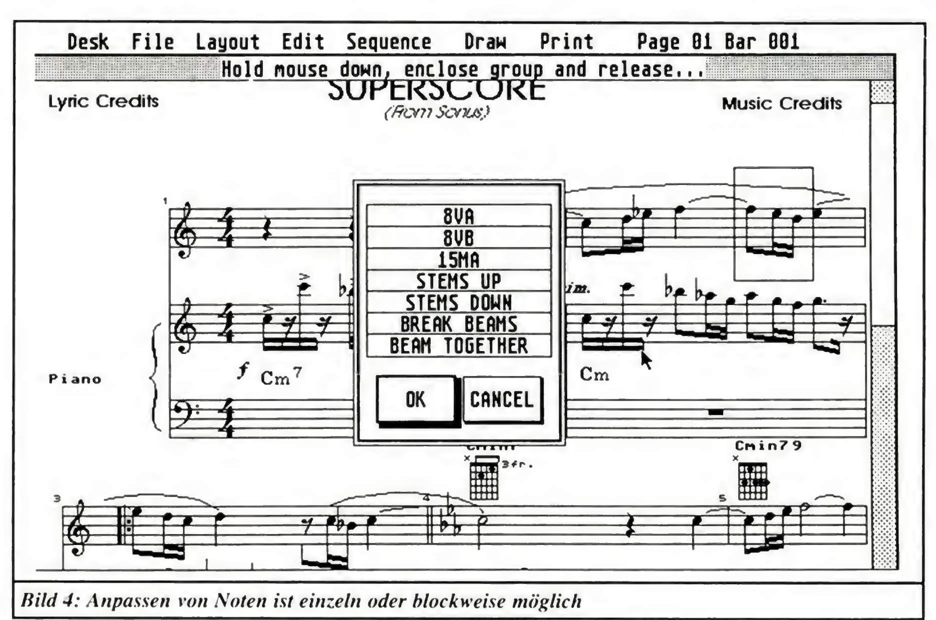
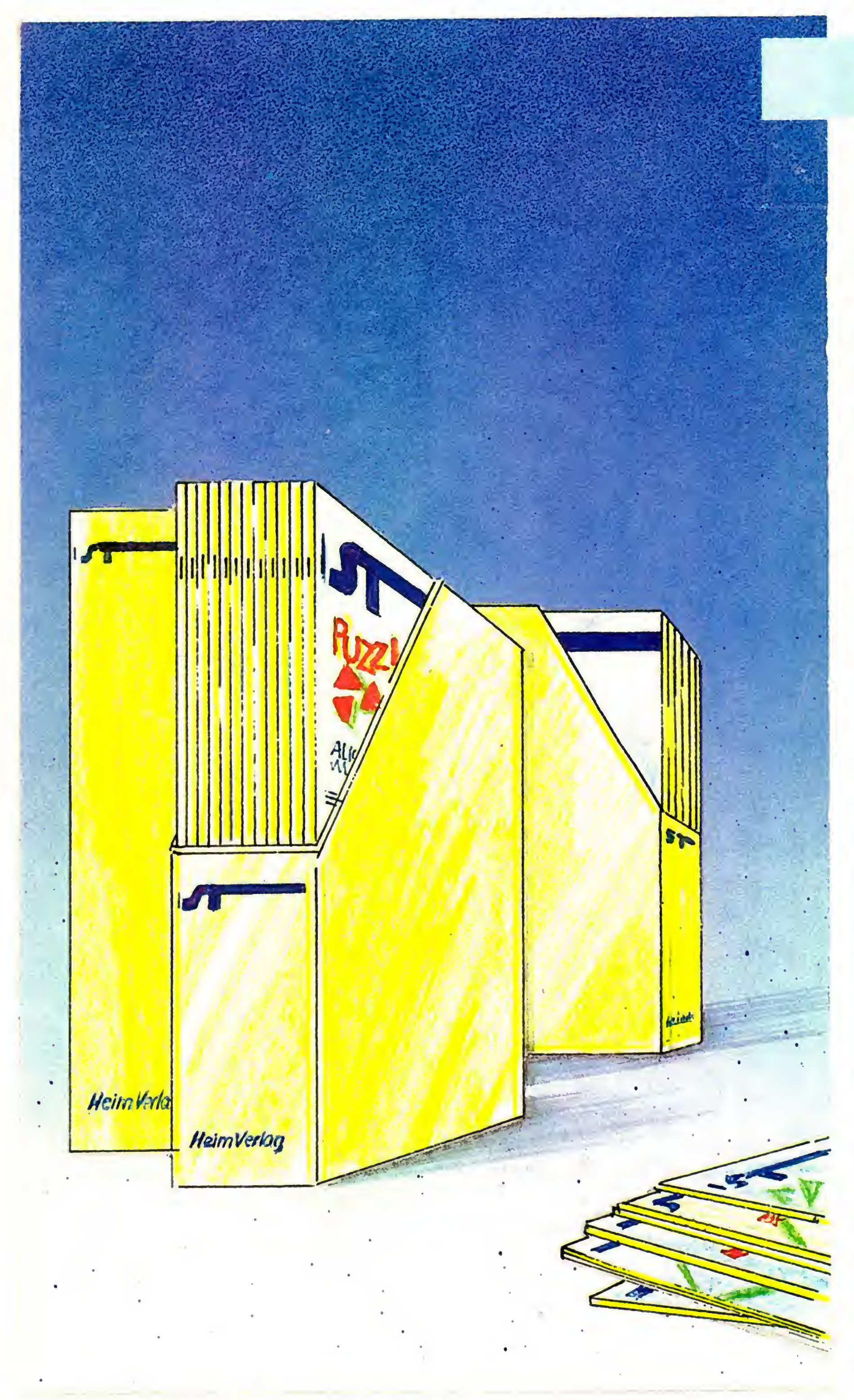


Bild 4: Anpassen von Noten ist einzeln oder blockweise möglich

BRINGEN SIE ORDNUNG IN IHRE ATARI ST ZEITSCHRIFTEN-SAMMLUNG MIT DEN PRAKTISCHEN SAMMEL-BOXEN!



VERGESSEN SIE

- Gestapelte, verrutschte und gekippte Zeitschriften
- Mühseliges Suchen nach einer bestimmten Ausgabe

NUTZEN SIE DIE VORTEILE DER NEUEN SAMMELBOXEN

- Übersicht
- Schneller und leichter Zugriff auf gewünschte Ausgaben
- Platzsparende und geschützte Aufbewahrung Ihrer Sammlung
- Ihre ST-COMPUTER-Zeitschriften werden zu einem geordneten und wertvollen Nachschlagewerk

BESTELLEN SIE:

- mit untenstehendem Bestellabschnitt
- Eine Box kostet DM 14,– zuzügl. einer Versandkostenpauschale von DM 3,–, unabhängig von der bestellten Stückzahl
- Die Auslieferung Ihrer Bestellung wird nur gegen Vorauszahlung durchgeführt
- Zahlen Sie bitte mit Scheck oder durch Vorab-Überweisung unter genauer Angabe Ihrer Wünsche.
- Die Auslieferung erfolgt sofort nach Zahlungseingang

ÜBRIGENS:

- Mit den neuen ST-COMPUTER-Sammelboxen oder mit einem Geschenk-Abonnement von ST-COMPUTER machen Sie Freunden und Bekannten große Freude

unverbindlich empfohlener Verkaufspreis

Benutzen Sie untenstehenden Bestellabschnitt und die Bestellkarten in der Zeitschrift.

Heim Verlag
Heidelberger Landstr. 194
6100 Darmstadt-Eberstadt
Telefon 06151-56057

BESTELLCOUPON

Bitte senden Sie mir ________ Stück Sammelboxen ST COMPUTER zum Preis von DM 14,– je Stück zuzüglich DM 3,– Versandkosten unabhängig von der Bestellmenge.

Zahlung: ☐ per Nachnahme ☐ Scheck liegt bei ☐ per Vorausüberweisung

Name, Vorname ________________________________

PLZ, Ort ________________ Straße, Nr. ________________

Datum ________________ Unterschrift ________________

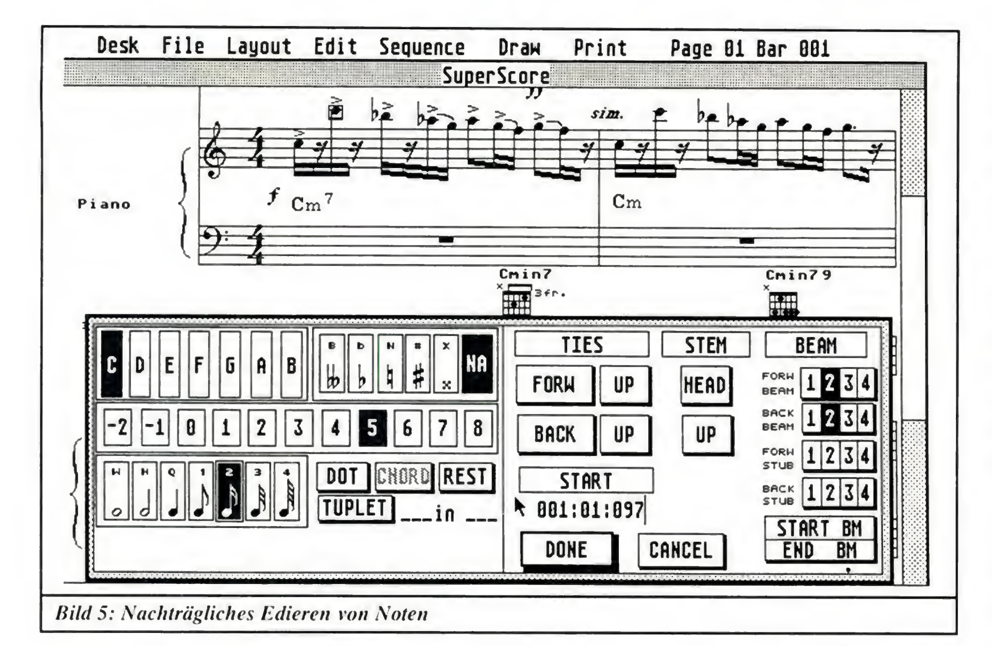


Bild 5: Nachträgliches Edieren von Noten

gehören, verschoben werden. Das geht zum größten Teil auch sehr komfortabel vor sich: Entweder man klickt eine einzelne Note an. Dann erscheint unter der Menüzeile eine Eingabezeile, in der man mittels Tastendruck auf der ST-Tastatur eine Option auswählen kann, z.B. Löschen, Verschieben oder Edieren. Wählt man letzteres, so erscheint eine Dialogbox, in der man so ziemlich alle Parameter einer Note verändern kann. Änderungen der Tonhöhe sind beim Abspielen auch sofort hörbar.

Der zweite Weg, Noten oder auch Gruppen von Noten zu verändern, benutzt nur die Maus: einfach ein paar Noten mit dem Gummirechteck einfangen. Dann erscheint eine kleine Dialogbox, in der Sie Verschiedenes wählen können: Noten um ein oder zwei Oktaven versetzen, Notenhälse nach oben oder nach unten oder Balken zusammenfassen oder auseinandernehmen.

Alle diese Edit-Kommandos werden auf der Seite nur angedeutet; man muß ein Redraw-Kommando aufrufen, mit dem man die Seite neu zeichnen kann.

Leider gibt es bei den Kommandos, so schön und nützlich sie im Prinzip sind, noch ein kleines Problem: Das Programm macht noch zu viele Fehler. Zwar stürzt es praktisch nicht ab (die aktuelle Version ist während des ganzen Testes nicht abgestürzt), aber die Ergebnisse dieser notenbezogenen Edierkommandos, speziell, soweit sie die Notenhälse und Fähnchen betreffen, sind unbefriedigend. An dieser Stelle müssen dringend ein paar Bugs entfernt werden, sonst helfen die tollen Möglichkeiten wenig.

Zwei Optionen helfen in diesem Stadium bei der Verbesserung des Layouts: Balken können schräg oder horizontal gesetzt werden. Auf den meisten Druckern sehen gerade Balken natürlich besser aus, weil man keine Treppchen sieht, aber über Geschmack läßt sich nicht streiten.

Sehr nützlich ist die Möglichkeit, auch Pausen in Balken einzubeziehen. Komplizierte rhythmische Figuren werden so erheblich übersichtlicher.

Wenn die Funktionen dieser Layout-Ebene richtig funktionieren, sind den Möglichkeiten kaum Grenzen gesetzt. Es ist mir zum Beispiel bisher mit keinem anderen Notendruckprogramm auf dem ST gelungen, verhältnismäßig mühelos einen Ausschnitt aus einem Gitarrentrio mit komplizierter Polyphonie zu setzen; mit SuperScore funktionierte es wunderbar, bis auf einige Stellen, an denen programminterne Bugs mir einen Strich durch die Rechnung machten (Da malt das Programm doch tatsächlich an einer bestimmten Stelle die Balken einfach wild in die Noten statt an die Notenhälse. Aber immer nur an bestimmten Stellen. Dabei geht es sonst so toll...). Liebe Programmierer, bitte, bitte, beseitigt schnell die Fehler des Noteneditors!

Auch Taktstriche kann man verschieben und löschen, wenn sie im Weg sind. Natürlich läßt sich auch die Art der Taktstriche verändern: Wiederholungszeichen und Doppelstriche sind genauso möglich wie verschiedene Endungen nach Wiederholungen (maximal drei: 1., 2. und 3. Haus).

Ganze Takte können kopiert, eingefügt oder gelöscht werden, auch das Verlängern eines Taktes ist möglich, um z.B. einen Taktwechsel ohne Veränderung der folgenden Noten zu produzieren.

Natürlich ist es auch möglich, Texte unter einem System zu plazieren. SuperScore hilft bei der Plazierung der Silben, indem es auf Kommando den Textcursor an einer bestimmten Stelle positioniert. Das Textformat ist wie bei den Titeln frei wählbar. Außer dieser speziellen Methode, an die Noten gebundene Texte zu produzieren, kann man auch freien Text auf der Seite verteilen. Damit sollte man jedoch solange warten, bis garantiert keine Änderungen am 'Vertical Spacing' mehr zu erwarten sind, sonst stimmt die Positionierung nicht mehr.

Ein besonderes Bonbon für die Arrangeure von Jazzbands, Liederbüchern oder jeder anderen Literatur, die sich mit Gitarrenakkorden beschäftigt, ist die Akkordsymbol-Funktion. Es gibt kleine und große Griffsymbole sowie eine Textumschrift. Man muß die Akkorde auch nicht erst selbst eingeben, sondern kann sie aus einer riesigen Akkordbibliothek, die sich ebenfalls als ASCII-File auf Diskette befindet, auswählen. Dazu muß man nur den ungefähren Akkordnamen eingeben, z.B. Csus4, und SuperScore sucht den Akkord aus der Bibliothek, der dem gewünschten am nächsten kommt (nach lexikalischen Gesichtspunkten natürlich). Wenn man Sonderwünsche hat, braucht man diese nur in die Bibliothek einzutragen. Die Positionierung der Symbole wird durch eine Art Snap-Funktion erleichtert, die es ermöglicht, jeweils eine Bewegungsrichtung der Maus zu blockieren, so daß man den Cursor nur vertikal oder nur horizontal bewegen kann. Diese Akkordsymbol-Funktion kann sich wirklich sehen lassen.

Den letzten Schliff erhält die Partitur dann durch musikalische Sonderzeichen, die in allen Formen und Größen zur Verfügung stehen. SuperScores Sonderzeichen-Bibliothek ist recht umfangreich, sie enthält z.B. auch Flageolett- und Schlagzeug-Noten. Sollten Sie aber tatsächlich auch hier Sonderwünsche haben, nichts leichter als das:

Einmal können Sie mit einem Zeichenprogramm wie DEGAS ELITE das entsprechende Sonderzeichen oder auch Ihre Unterschrift oder was auch immer als Grafikblock zeichnen und abspeichern und diesen Grafikblock in SuperScore einladen und dort wie eines der eingebauten Sonderzeichen verwenden. Es kommt aber noch viel besser (Freunde Neuer Musik bitte herhören !):

Ein recht umfangreiches, objektorientier-

tes Zeichenprogramm ist gleich eingebaut, Sie können beliebig breite Linien, z.B. für Cluster-Notation zeichnen, oder besondere n-Tolen und Bindebögen oder, was Ihnen eben einfällt, um Ihre Noten zu verschönern. Jede einzelne Linie, die Sie auf diese Art zeichnen, können Sie auch wieder löschen, wie sich das eben für ein objektorientiertes Zeichenprogramm gehört. Ich glaube, auf diese Art kann man die eigenen Werke wirklich extrem frei gestalten.

Leider ist auch dieser Teil des Programmes nicht ganz frei von Fehlern: Nach einem Redraw-Kommando verschwinden manchmal Sonderzeichen vom Bildschirm, d.h. sie sind nicht mehr sichtbar, aber offensichtlich noch vorhanden, denn löschen kann man sie noch. Ärgerlich, aber hoffentlich bald behoben.

Ein paar Worte zur Bedienung

SuperScore ist kein Weltmeister in Sachen Geschwindigkeit, es zeichnet bedächtig, ein Bildschirm-Redraw dauert einige Sekunden. Das Programm ist allerdings so konzipiert, daß man mit etwas Mitdenken mit sehr wenigen Redraws auskommt. Überhaupt sollte man SuperScore mit Bedacht nutzen: Bevor man ein Layout macht, sollte man eine recht klare Vorstellung vom Aussehen des Endproduktes haben, sonst kann sich die Arbeit in die Länge ziehen. Wenn die Fehler nicht wären, könnte man sehr gut mit dem Programm arbeiten; unter Berücksichtigung der Fehler ist die Arbeit an komplexen, vor allem polyphonen Layouts stark gehandicapt. Ich persönlich hoffe für meine eigene Arbeit stark auf ein Update, das die Fehler korrigiert, denn in Sachen Möglichkeiten beim Notensatz gibt es bisher keinen Konkurrenten auf dem ST für SuperScore; außer dem Copiisten von Dr. T's, dem es für meinen Geschmack an Bedienungskomfort mangelt. Es bleibt aber dabei: Die Fehler müssen raus, und zwar schnell.

Notensatz ist, was die Bedieneroberfläche betrifft, ein schwieriges Problem. SuperScore bietet in vielen Punkten komfortable Lösungen dafür.

Ein nicht ganz unwichtiger Punkt für ein Notendruckprogramm ist die Druckqualität. Hier macht sich SuperScore leider eine Eigenart vieler amerikanischer Programme zu eigen: Viele Druckertreiber stehen zur Verfügung, nämlich so ziemlich alles, was es für GEM (DEGAS

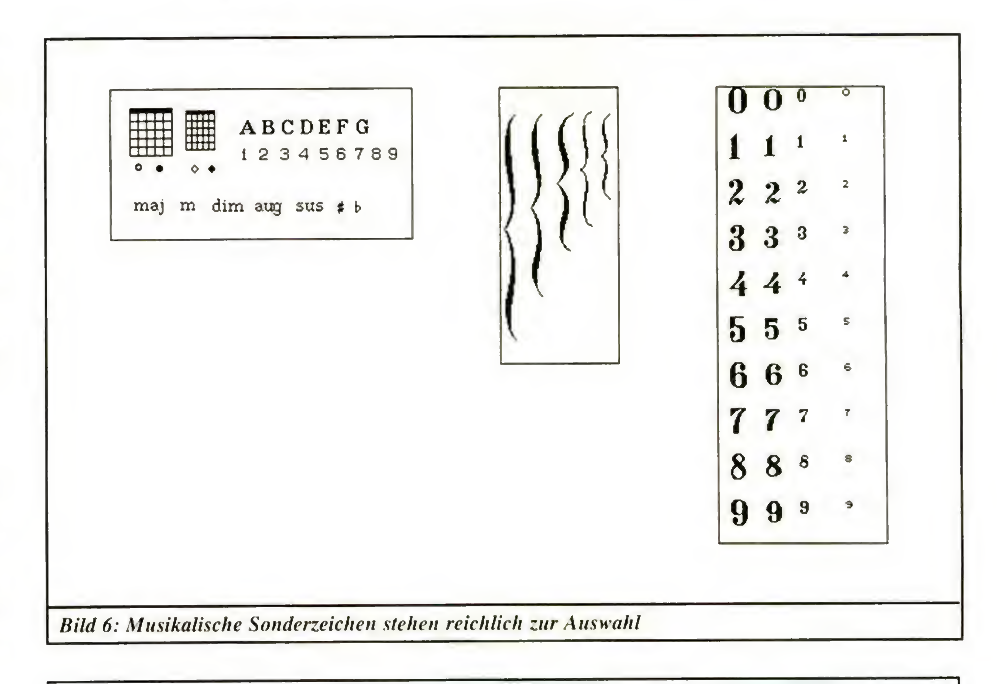


Bild 6: Musikalische Sonderzeichen stehen reichlich zur Auswahl

ST-Computer testet
SuperScore
von Sonus
Texter-Belobigung
desgleichen fuer den Komponisten
Allegro
Fagott
Klavier
Schlagzeug
© ST Computer Seite 1 August/September 1988

Bild 7: Ein Probeausdruck mit SuperScore

ELITE, amerikanische PD, ANTIC usw.) gibt. Diese Treiber lassen sich per Menübefehl einladen, als Default gibt es einen Epson-kompatiblen Treiber.

Leider läßt sich mit diesen Treibern keine besonders gute Qualität erreichen, auch nicht auf 24-Nadel-Druckern. Mit Programmen wie C-Labs Notator oder Steinbergs Masterscore, ja selbst dem kleinen EZ-Score plus von Hybrid Arts kann SuperScore leider noch nicht mithalten.

Allerdings ist für die nächste Zeit ein Laser-Treiber angekündigt, der dann auch Postscript-fähig sein soll. ADOBEs Sonata-Font soll verwendet werden, und der ist wirklich sehr schön. Auch auf dem 24-Nadel-Sektor soll sich mit dem neuen Treiber einiges tun. Warten wir's ab.

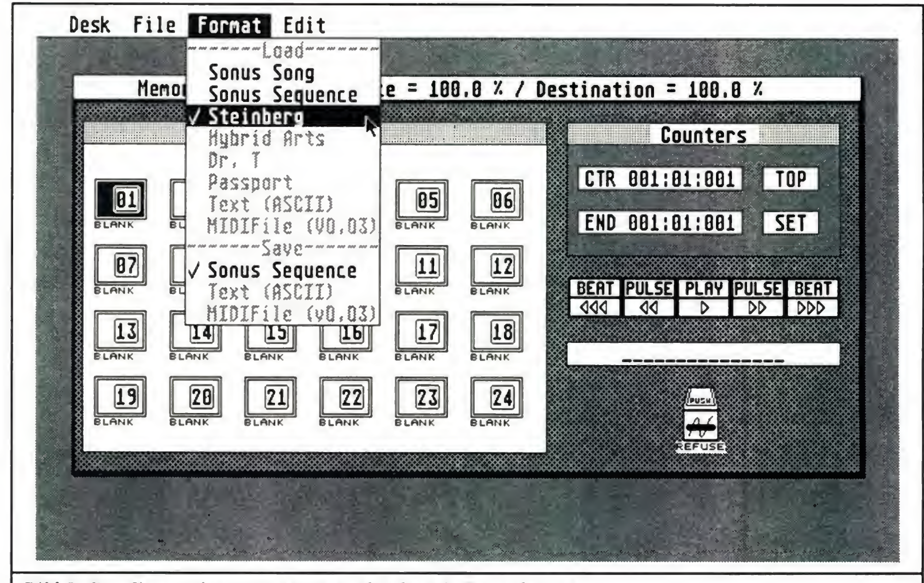


Bild 8: Das Konvertierungsprogramm für fremde Datenformate

Symbole, die mit dem eingebauten Zeichenprogramm gemalt wurden, sollen dann übrigens auch in höherer Auflösung mitgedruckt werden. Aber das ist Zukunftsmusik.

Zum Schluß sei noch erwähnt, daß ein kleines Hilfsprogramm im Lieferumfang enthalten ist, mit dem es möglich ist, Sequenzen von fremden Sequencern in das SuperScore-Format zu konvertieren. Im Augenblick wird nur das Steinberg TwentyFour-Format, basierend auf Version 2.0, unterstützt, bei meinen Versuchen ließen sich aber auch Files der Versionen 2.1 und 3.0 laden, wenn auch nicht immer problemlos. In der Menüleiste sind bereits einige andere verbreitete Formate vorgesehen, so z.B. das Midi-Standard-Format.

SuperScore hinterläßt einen etwas zwiespältigen Eindruck: Das Programm hat ganz hervorragende Anlagen, die Möglichkeiten, die an sich vorhanden sind, sind einfach toll. Leider aber sorgen ein paar kleine Fehler irgendwo im Programm dafür, daß man nicht allzuviel mit den großartigen Möglichkeiten anfangen kann, wenn man mehr tun will, als man auch mit herkömmlichen Notendruckprogrammen tun kann (mehr, als nur einfache Partituren mit homophonen Stimmen zu drucken). Es ist wirklich ärgerlich: Endlich mal ein Programm, das die Features bietet, die man dringend braucht, und dann ärgern einen lauter kleine Teufelchen. Aber immerhin, man kann schon damit arbeiten. Das nächste Update kommt bestimmt.

Der zweite Haken ist die Druckqualität. Auch und gerade hier muß sich einiges tun. Angekündigt ist es, aber noch warten wir. Schließlich sollte es ein neues und vor allem auch ein deutsches Handbuch geben. Das Preis-/Leistungsverhältnis ist mit einem Preis von DM 698,- auf jeden Fall in Ordnung. Für eine Kombination aus bodenständigem Sequencer und leistungsfähigem Noten-Layoutprogramm ist der Preis nicht hoch. Mit einem halbwegs fehlerfreien SuperScore könnte man sehr zufrieden sein. Hoffentlich wird es bald eine deutsche und eine korrigierte Version mit einer vernünftigen Anleitung geben.

CS

Bela Computer
Unterortstr. 23-25
6236 Eschborn
Tel.: 06196/481944

ENDE

WENN DER ATARI ST MIT DER GABI ANBANDELT

Teil 2



In der heutigen Ausgabe wollen wir uns weiter mit der Verbindung von ATARI ST und TA-Gabriele 9009 bzw. kompatiblen Schreibmaschinen beschäftigen. Doch bevor wir uns in das Reich der Bits und Bytes begeben, möchte ich die Aufgabenstellung definieren:

Gesucht wird ein Programm, das im Auto-Ordner mitgebootet wird und sich resistent im Speicher hält. Dieses Programm "verbiegt" die Druckerausgabe auf eine Routine, die die spezielle Aussteuerung der Schreibmaschine entsprechend der in der letzten Ausgabe aufgeführten Befehlssequenzen realisiert.

Außerdem installiert es eine Interruptroutine, die dafür sorgt, daß die Schreibmaschine über die entsprechenden Tasten von der Schreibmaschinentastatur aus zwischen den Betriebszuständen ONLINE und OFFLINE umgeschalten werden kann.

Das Programm sorgt auch für die Verwaltung der Druckschlittenposition und für die Realisierung einiger spezieller Funktionen.

Alle diese nachfolgend aufgelisteten Funktionen werden durch Codes bzw. Sequenzen gesteuert, die vollkompatibel zur Original-Interfacebox von TA sind. Dadurch ist es möglich, einen entsprechenden Druckertreiber mitzuverwenden. Es ist durch Änderungen im Programm aber auch möglich, eine Anpassung an beispielweise EPSON-Sequenzen zu erreichen.

ASCII-Code-Befehle

Folgende Steuerzeichen innerhalb des ASCII-Codes werden von der Druckroutine erkannt und realisiert:

- **08H : Backspace**: Der Druckschlitten wird um eine Schreibteilung nach links bewegt.

- **0AH : Linefeed**: Dieser Code löst das Weiterschalten der Schreibmaschine um eine Zeile aus.

- **0CH : Form Feed**: Dieser Befehl führt üblicherweise dazu, daß das Papier zum Anfang der nächsten Seite transportiert wird (Endlospapier). Da die Schreibmaschine allerdings nur Einzelblattverarbeitung erlaubt, wurde die Funktion im Programm so realisiert, daß die Schreibmaschine beim Auftreten des Befehls in den OFFLINE-Modus geschaltet wird. Sie wartet dann, bis ein neues Blatt eingespannt und über die ONLINE-Taste auf der Schreibmaschinentastatur das Weiterdrucken veranlaßt wurde.

- **0DH : Carriage Return**: Der Druckschlitten wird zum linken Rand zurückbewegt.

- **1BH: ESCAPE**: Erstes Byte einer folgenden Befehlssequenz.

- **20H: Space**: Leerzeichen entsprechend der eingestellten Schreibteilung. Um eine Geschwindigkeits-Optimierung bei mehreren aufeinanderfolgenden Leerzeichen zu erreichen, werden diese nicht einzeln ausgeführt, sondern zu einer schnelleren direkten Bewegung zusammengefaßt.

Andere Kontrollbytes, die im Bereich der ASCII-Codes bis 20H (z.B. DC1/3 oder HT) liegen, werden üblicherweise nicht benutzt und deshalb von unserem Programm einfach unterdrückt.

ESCAPE-Sequenzen

Unter ESC-Sequenz versteht man eine Reihenfolge von Steuerbefehlen, die mit einem ESC-Zeichen (=1BH) beginnen und druckerspezifische Funktionen (z.B. Veränderung der Schreibteilung) steuern. Leider sind diese Sequenzen für die verschiedenen Drucker nicht standardisiert. Unser Programm richtet sich nach den von der Original-Interfacebox vorgegebenen ESC-Sequenzen. Allerdings wurde nur eine Untermenge der von der IFD1 vorgegebenen Sequenzen realisiert, jedoch genügen diese, um eine leistungsfähige Funktionalität zu erreichen.

- **ESC 9 (= 1BH + 39H): Linken Rand setzen**
Erläuterung: Der linke Rand wird auf die Position gesetzt, auf der gerade der Druckschlitten steht.

- **ESC D (= 1BH + 44H): Hochstellen**
Erläuterung: Die Schreibwalze wird um

eine halbe Grundzeile zurückgedreht, wodurch ein Hochstellen der Zeichen auf dem Papier erfolgt.

- ESC U (= 1BH + 55H): Tiefstellen
Erläuterung: Die Schreibwalze wird um eine halbe Grundzeile vorgedreht, wodurch ein Tiefstellen der Zeichen auf dem Papier erfolgt.

- ESC LF (= 1BH + OAH): Linefeed rückwärts
Erläuterung: Die Schreibwalze wird entsprechend dem eingestellten Zeilenabstand eine Zeile zurückbewegt.

- ESC W (= 1BH + 57H): Fettschrift einschalten
Erläuterung: Alle anschließenden Zeichen werden in Fettdruck dargestellt. Dies erfolgt dadurch, daß jedes Zeichen zweimal abgeschlagen und zwischen den Anschlägen eine minimale Horizontalbewegung (1/120") ausgeführt.

- ESC & (= 1BH + 26H): Fettschrift ausschalten
Erläuterung: Alle nachfolgenden Zeichen werden wieder ohne Fettdruck abgedruckt.

- ESC E (= 1BH + 45H): Unterstreichen einschalten
Erläuterung: Bei allen nachfolgenden Zeichen wird zusätzlich zum Zeichenabdruck auch ein Unterstreichstrich ("_") abgedruckt.

- ESC R (= 1BH + 52H): Unterstreichen ausschalten
Erläuterung: Alle nachfolgenden Zeichen werden wieder ohne Unterstreichen abgedruckt.

- ESC US x (= 1BH + 0BH + x): Schreibteilung setzen auf x-Schritte
Erläuterung: Die Schreibteilung wird auf 12 (12 Zeichen pro Zoll) vorbesetzt. Mit dieser Sequenz ist es möglich, die Schreibteilung im Bereich von 1/120" bis 255/120"-Schritten einzustellen.

Die Sequenzdefinition der Interface-Box sagt aus unerklärlichen Gründen, daß als Schrittzahl (x + 1) übergeben werden muß.

Nachfolgend ist die Zuordnung der Schrittzahlen zu den gebräuchlichsten Schreibteilungen aufgeführt:

Schreibteilung	Schrittzahl X
10	13
12	11
15	9
6	23

```
**********************************************************************
*                                                                    *
*    Nachbildung der INTERFACEBOX IFD1 durch Software im ST          *
*    (Einsatz des deutschen Typenrades Grp. 01)                      *
*                                                                    *
*    Diese Datei enthält Tabellen bezüglich der CODE SEQUENZEN und   *
*    des Zeichensatzes. Dadurch ist es möglich, die verschiedenen    *
*    Funktionen und den Zeichensatz anzupassen.                      *
*                                                                    *
**********************************************************************

*
* Name of printer
* ===============
*
IFD 1 - Emulator
*
* Miscellaneous configurable variables
* ====================================
*
* 1: printer type, 0=dot matrix, 1=daisy wheel
*    Note if printer type is 0 the following 4 variables are never used.
* 2: unit width of one character
* 3: unit height of one line
* 4: Approximate middle of carriage after formfeed
* 5: Carriage shift for bold overstrike
*
0,0,0,0,0,0

*
* Printer characteristics
* =======================
*
* This table specifies the printer command sequences.
* If the top bit of a code is set, then this indicates the position
* of a parameter passed to the printer. The code whose top bit is set
* in this table is added to the parameter passed before being sent to the
* printer. It is not used in all command sequences, only in those where
* the printer requires  a variable value such as the length of a vertical
* tab.
*
* 0              * Character width
1, D, A          * Linefeed WITH return
* 2              * Forward print
* 3              * Reverse print
* 4              * Vertical tab to line
* 5              * Absolute horizontal tab
6                * Draft bold on
7                * Draft bold off
8, 1B, 57        * Near Letter Quality (NLQ) bold on
9, 1B, 26        * NLQ bold off
A                * Draft italic on
B                * Draft italic off
C                * NLQ italic on
D                * NLQ italic off
E                * Draft light on
F                * Draft light off
* 10, 1B, 78, 0  * NLQ light on
* 11, 1B, 78, 1  * NLQ light off
12, 1B, 44       * Draft superscript on
13, 1B, 55       * Draft superscript off
14, 1B, 44       * NLQ superscript on
15, 1B, 55       * NLQ superscript off
16, 1B, 55       * Draft subscript on
17, 1B, 44       * Draft subscript off
18, 1B, 55       * NLQ subscript on
19, 1B, 44       * NLQ subscript off
1A
1B                * Draft underline off
1C, 1B, 45        * NLQ underline on
1D, 1B, 52        * NLQ underline off
1E, 0C,oD         * Formfeed
1F, 12            * Horizontal initialisation
20,0D             * Termination: printer reset
21,0D             * Termination: printer reset
22,08                           * Backspace
23,13                           * Carriage Return
* 24                            * Form length in lines
* 25                            * Set line feed to N units
* 26                            * Start graphics (mode 0)
* 27                            * Reserved
28,1B,1F,0D                     * Draft Pica      (10   cpi)
29,1B,1F,0D                     * NLQ   Pica      (10   cpi)
2A,1B,1F,0B                     * Draft Elite     (12   cpi)
```

Tabelle geht weiter...

```
2B,1B,1F,0B                    * NLQ   Elite      (12   cpi)
2C,1B,1F,9                     * Draft Condensed (15   cpi)
2D,1B,1F,9                     * NLQ   Condensed (15   cpi)
2E,1B,1F,15                    * Draft Expanded  ( 6   cpi)
2F,1B,1F,15                    * NLQ   Expanded  ( 6   cpi)
* 30                           * Black   ink
* 31                           * Magenta ink
* 32                           * Cyan    ink
* 33                           * Yellow  ink
0                              * NULL termination byte

*
* Translation Table
* =================
*
* Diese Tabelle übersetzt einfache ATARI Eingangscodes in vielfache
* Druckercodes, dies ermöglicht den Entwurf eines erweiterten Zeichen-
* satzes usw. .
*
* Da der Zeichensatz  schon vom Steuerprogramm verwirklicht wird, ist es
* hier nicht nötig, nochmals eine Übersetzung durchzuführen
```

Abb. 1: Tabelle der CODE-SEQUENZEN

- ESC RS x (= 1BH + 1EH + x): **Zeilenabstand setzen** auf (x + 1) Schritte Erläuterung: Der Zeilenabstand ist auf die genormte Grundzeile von 1/6 " voreingestellt. Durch die Sequenz ist es möglich, den Zeilenabstand zwischen 1/96" und 255/96" einzustellen. Für die Grundzeile ist als x beispielsweise der Wert 17 einzusetzen.

Absichtlich nicht realisiert wurden die ESC-Sequenzen für bidirektionalen Druck (Druck links - rechts - links usw.), da die Schreibmaschine Gabriele 9009 zwar prinzipiell von rechts nach links drucken kann, jedoch aus internen Gründen nach jedem Schritt nach links eine seltsame Rüttelbewegung ausführt. Damit wird der durch den bidirektionellen Druck gewonnene Zeitvorteil mehr als aufgebraucht.

Nicht verwirklicht wurde auch die Funktion Proportionaldruck. Proportionaldruck bedeutet, daß jedem Zeichen ein spezifischer Horizontalabstand zugeordnet ist; der Abstand zwischen "i" ist somit viel schmaler als zwischen "M". Wir haben auf die Realisierung der Funktion verzichtet, da diese nur in Zusammenarbeit mit speziellen Typenrädern ansprechende Ergebnisse bringt und andererseits großen Aufwand im Programm sowie in entsprechenden Treibern bedeutet hätte.

Das nachfolgende aufgelistete Programm wurde in Assembler erstellt, um möglichst "maschinennah" arbeiten zu können. Die enthaltenen Kommentare sollten genügen, die verschiedenen Programmabschnitte nachvollziehbar zu machen. Es wird somit auch möglich sein, bei Bedarf kleinere Modifikationen einzubringen.

Da der ATARI ST über die serielle Schnittstelle während des Hochlaufs oft Undefiniertes aussendet, ist es sinnvoll, die angeschlossene Schreibmaschine erst nach abgeschlossenem Boot-Vorgang einzuschalten. Denn empfängt die Schreibmaschine Zeichen, mit denen sie nichts anfangen kann, ignoriert sie diese nicht immer, sondern stürzt manchmal auch ab, was nur durch erneutes Aus-/Einschalten der Schreibmaschine behoben werden kann.

Bevor man mit dem Drucken beginnen kann, muß man die Schreibmaschine durch Betätigen der ONL-Taste (oder entsprechende Tastenkombination) auf ONLINE schalten.

Obwohl die serielle Verbindung physikalisch über die V.24-Schnittstelle erfolgt,ist als logische **Schnittstelle** die voreingestellte **Parallelschnittstelle** zu verwenden. Für WORDPLUS-Benutzer habe ich noch einen kleinen Treiber erstellt, der die vorgenannten ESC-Sequenzen beim Ausdrucken automatisch ansteuert.

Eine Code-Umsetzung der Schreibzeichen ist nicht erforderlich, da unser "IFD-EMULATOR" schon alle druckfähigen Zeichen ansteuert.

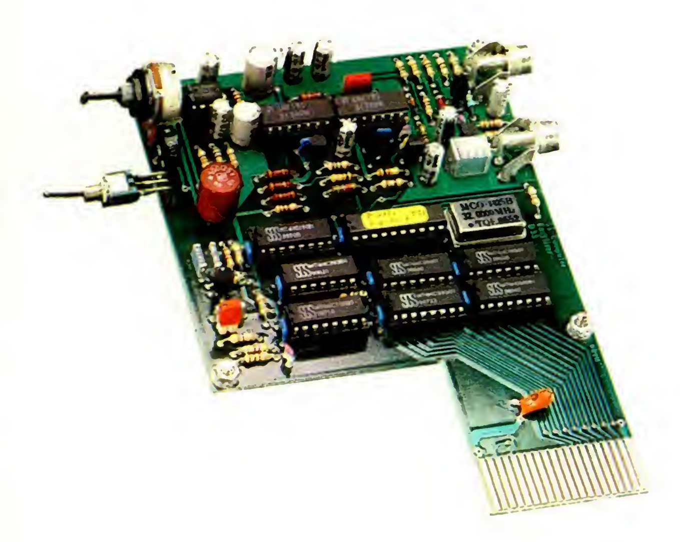
Wilfried Keller

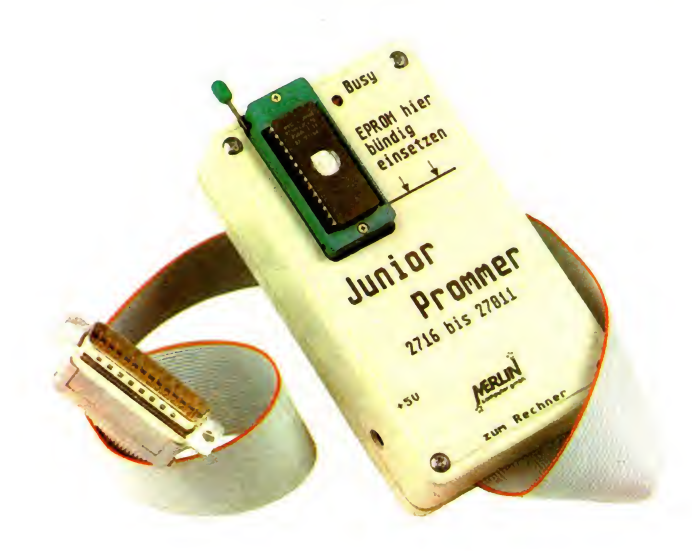
Listing folgt nächste Seite...

```
 1: *******************************************************
 2: *                                                     *
 3: * IFD 1  -  EMULATOR für ATARI ST                     *
 4: *                                                     *
 5: * Diese Routine verbiegt d.Druckerausgabe a.ein Kon-  *
 6: * vertierprogramm, das d.ASCII-Code i.f. die Schreib- *
 7: * maschine Gabriele 9009 verständl.Steuerbefehle um-  *
 8: * setzt. Die Kommunikation m.d.Schreibmaschine erfolgt*
 9: * o.Spooler-Pufferung ü.d.serielle V.24-Verbindung.   *
10: *                                                     *
11: *******************************************************
12:
13:
14:         move.b #16,zeile
15:         move.b #10,teilung
16:         move.l  4(sp),a0    ; Groesse des Programms
17:         move.l  #$100,d6    ; berechnen
18:         add.l   12(a0),d6
19:         add.l   20(a0),d6
20:         add.l   28(a0),d6
21:         move.l d6,-(sp)     ; Ergebnis auf Stack
                                  zwischenlagern
22:
23:         clr.l -(sp)         ; In SUPER-Modus schalten
24:         move.w #$20,-(sp)
25:         trap #1
26:         add.l #6,sp
27:         move.l d0,-(sp)     ; Alten Stack merken
28:
29:         move.l  #trap13,-(sp)   ; TRAP13 "verbiegen"
30:         move    #45,-(sp)
31:         move    #5,-(sp)
32:         trap    #13
33:         addq.l  #8,sp
34:         move.l  d0,tr_save ; alte Sprungadresse merken
35:
36:
37:         move.l #em_int,-(sp) ; Interruptroutine
                                   installieren,
38:         move.w #12,-(sp)     ; die ausgelöst wird,
                                   wenn von der SM ein
39:         move #13,-(sp)       ; Zeichen empfangen wurde
40:         trap #14             ; -> Offline - Online
41:         addq.l #8,sp
42:
43:         bsr init             ; Intialisierungsroutine
44:         move.l (sp)+,d0      ; In USER-Modus
                                   zurueckschalten
45:         move.l d0,-(sp)
46:         move.w #$20,-(sp)
47:         trap #1
48:         addq.l #6,sp
49:
50:         move.l (sp)+,d6  ; Programmgroesse zurückholen
51:
52:         clr     -(sp)    ; Programm im Speicher halten
53:         move.l  d6,-(sp)
54:         move    #$31,-(sp)
55:         trap    #1
56:
57:
58: trap13: move.l  sp,a2    ; neue Trap 13-Routine
59:         btst    #5,(sp)  ; jetzt abprüfen, ob
60:         bne     super    ; Druckerangesprochen ist
61:         move.l  usp,a2
62:         subq    #6,a2
63: super:  cmp     #3,6(a2)
64:         bne     normal   ; wenn nicht -> alte Routine
65:         cmp     #0,8(a2)
66:         bne     normal
67:         movem.l d1-d7/a0-a6,-(sp)
68:         move.l  $4a2,a1
69:         move    (sp)+,-(a1)
70:         move.l  (sp)+,-(a1)
71:         move.l  a1,$4a2
72:                          ; Drucker ist gemeint auszu-
73:         move    10(a2),d1 ; gebendes Zeichen nach d1
74:         move #$2700,sr
75:         cmp.w   #0,esc_flag ; Folgebyte in ESC-Sequenz?
76:         bne     esc_folg    ; Ja -> verzweigen
77:         cmp.b   #$20,d1     ; Druckbares Zeichen?
78:         bls     so_zei      ; Nein -> verzweigen
79:         cmp.w #0,step_li    ; Sind noch Leerschritte
                                  nachzuholen?
80:         bne spa_nach        ; Ja -> verzweigen
81: drck:   lea  zei_satz,a2    ; Adresse f.Tabelle nach a1
82:         and.l #$FFFF,d1
83:         sub.w #$21,d1       ; Offset von ASCII-Code
                                  subtrahieren
84:         lsl.w #1,d1         ; Rest mit 2 multiplizieren
85:         add.l  d1,a2        ; Als Offset zu Tabellenzeige
                                  addieren
86:         cmp.w #0,son_flag   ; Ist eine Sonderfunktion
                                  eingeschaltet?
87:         bne so_druck        ; Ja -> verzweigen
88: narma:  move.b (a2)+,d1     ; Typenrad-Code aus Tabelle
89:         bsr print           ; holen und an SM senden
90:         move.b (a2),d1      ; Abschlag-Code aus Tabelle
91:         bsr print           ; holen und an SM senden
92: gedr:   move.b teilung,d0
93:         and.w #$FF,d0
94:         add.w d0,spalte     ; Spaltenzaehler inkrem.
95:         moveq #-1,d0        ; kein Fehler aufgetreten
96:         bra exit            ; Aussprung
97:
98: spa_nach:                   ; Umwandeln von vorher
                                  nicht ausgedruckten
99:         move.l d1,-(sp)     ; Leerschritten in
                                  Direktbewegung
100:        move.w step_li,d1   ; Anzahl der auszuführenden
                                  Leerschritte
101:        lsr #8,d1
102:        bset #7,d1          ; Befehl für SM bilden
103:        bset #6,d1
104:        bclr #5,d1
105:        bclr #4,d1
106:        bsr print
107:        move.w step_li,d1   ; High_byte für direkte
                                  Bewegung
108:        bsr print
109:        move.w #0,step_li   ; Low_byte für direkte
                                  Bewegung
110:        move.l (sp)+,d1
111:        bra drck            ; zurück -> normal drucken
112:
113:
114: so_druck:
115:        move.l a2,a4
116:        btst #1,son_flag ; unterstreichen eingeschaltet?
117:        beq fett         ; Nein -> Weiter mit Fettdruck-
                               Abfrage
118:        move.b #$33,d1   ; Typenradcode für "_"
119:        bsr print
120:        move.b #$10,d1   ; Abschlagscode und
                               Totzeichenkennung
121:        bsr print
122: fett:  btst #2,son_flag  ; fett eingeschaltet?
123:        beq narma         ; Nein -> Normal weiter
124:        move.b (a2)+,d1 ; Typenradcode aus Tabelle holen
125:        bsr print
126:        move.b (a2),d1  ; Abschlagscode aus Tabelle holen
127:        bclr #7,d1      ; Horizontalbewegung unterdrücken
128:        bsr print
129:        move.b #$83,d1  ; Mikrostep (1/120") ausführen
130:        bsr print
131:        move.b #1,d1
132:        bsr print
133:        move.b (a2),d1  ; War es wirklich ein druckbares
                              Zeichen
134:        cmp.b #0,d1
135:        beq gedr        ; Nein -> Aussprung
136:        sub.l #1,a2     ; Ja -> Zeichen nochmal drucken
137:        bra narma
138:
139: esc_folg:
140:        cmp.w #1,esc_flag
141:        bne esc_3byt        ; 3. oder 4. Byte einer
                                  ESC-Folge
142:        move.w #0,esc_flag
143:        cmp.b #$39,d1       ; Linken Rand setzen?
144:        beq li_rand         ; Ja -> verzweigen
145:        cmp.b #$44,d1       ; Hochstellen?
146:        beq hochst          ; Ja -> verzweigen
147:        cmp.b #$55,d1       ; Tiefstellen?
148:        beq tiefst          ; Ja -> verzweigen
149:        cmp.b #$0A,d1       ; Zeilenrückschritt?
150:        beq lf_back         ; Ja -> verzweigen
151:        cmp.b #$45,d1       ; Unterstreichen ein?
152:        beq  unt_ein        ; Ja -> verzweigen
153:        cmp.b #$52,d1       ; Unterstreichen aus?
154:        beq unt_aus         ; Ja -> verzweigen
155:        cmp.b #$57,d1       ; Fettdruck ein?
```

Listing geht weiter...

Aus unserer Hardwareküche





Easytizer - der Videodigitizer ohne Geheimnisse

Mit dem Easytizer können Sie beliebige Videosignale von einer Schwarzweiß- oder Farb-Kamera, Videorecorder oder direkt vom Fernsehgerät (mit Composite Video Ausgang) digitalisieren und somit auf dem Bildschirm Ihres ATARI ST sichtbar machen.

Der Easytizer wird am Modul-Port des ST angeschlossen. Die Auflösung beträgt 800x600 Bildpunkte, so daß in mittlerer Auflösung 640x200 Bildpunkte in vier Graustufen dargestellt werden können. In dieser Betriebsart werden 12,5 Bilder in der Sekunde wiedergegeben. Im hochaufgelösten Modus werden 640x400 Bildpunkte in Schwarzweiß wiedergegeben.

Besondere Merkmale des Easytizers:

- Software vollständig in Assembler
- Abspeichern der Bilder im DEGAS-Format
- Von STAD und Sympatic Paint ansteuerbar
- Ein beliebiger Bildausschnitt kann in ein mit der Maus wählbares Format vergrößert und verkleinert werden
- Spiegeln eines Bildes in horizontaler und vertikaler Ebene
- Animation mit beliebig vielen Bildern möglich, nur durch die Kapazität des Rechners begrenzt (beim Mega ST4 über 100 Bilder)
- Eingebauter Druckertreiber für die mittlere Auflösung für NEC P6/P7 und EPSON oder Kompatible
- Wahlweise automatische oder manuelle Helligkeitseinstellung
- Schnappschuß

Lieferumfang:

1) Fertiggerät
   komplett aufgebaut und geprüft, inclusive Diskette mit der Easytizer Software und Bedienungsanleitung
2) Teilsatz
   Für Bastler liefern wir einen Teilsatz bestehend aus:
   - Doppelseitiger, elektronisch geprüfter Platine mit Lötstoplack und Bestückungskungsaufdruck sowie vergoldeten Anschlußkontakten
   - fertig programmiertes GAL 16V8
   - Quarzoszillatormodul 32 MHz
   - Diskette und Bedienungsanleitung.

Klein, kompakt und leistungsstark- der Junior Prommer

Der Junior Prommer programmiert alle gängigen EPROM-Typen, angefangen vom 2716 (2 KByte) bis zum modernen 27011 (1 MBit). Aber nicht nur EPROMs, sondern auch einige ROM- und EEPROM-Typen lassen sich lesen bzw. programmieren.

Zum Betrieb benötigt der Junior Prommer nur +5 Volt, die am Joystick-Port Ihres ATARI ST abgenommen werden, alle anderen Spannungen erzeugt die Elektronik des Junior Prommers. Die sehr komfortable Software, natürlich voll GEM unterstützt, erlaubt alle nur denkbaren Manipulationen.

Selbstverständlich läßt sich ein 16-Bit Word in ein High- und Low-Byte zerlegen. Fünf Programmieralgorithmen sorgen bei jedem EPROM-Typ für hohe Datensicherheit. Im eingebauten Hex/ASCII-Monitor läßt sich der Inhalt eines EPROMs blitzschnell durchsuchen oder auch ändern.

Alles dabei!

Bemerkenswert ist der Lieferumfang, so wird z.B. das Fertiggerät komplett aufgebaut und geprüft im Gehäuse mit allen Kabeln anschlußfertig geliefert. Auf der Diskette mit der Treibersoftware befinden sich noch RAM-Disk und ein Programm zum Erstellen von EPROM-Karten, ferner wird der Source-Code für Lese- bzw. Programmierroutinen mitgeliefert und last but not least ist im Bedienungshandbuch der Schaltplan abgedruckt.



Bestellcoupon Merlin Computer GmbH Industriestraße 26 6236 Eschborn Tel.: 06196/481811

Hiermit bestelle ich:

Name: ______________
Vorname: ______________
Straße: ______________
Ort: ______________
Unterschrift: ______________

	Artikel	Preis
☐	Junior Prommer (Fertiggerät wie beschrieben) nur	DM 189,00
☐	Leerplatine und Software (o. Bauteile)	DM 49,00
☐	Leergehäuse (gebohrt und bedruckt)	DM 39,90
☐	ROM-Karte 128 KByte bietet maximal 4 EPROMS Platz (fertigbestückt o. EPROMs)	DM 58,00
☐	Easytizer (Fertiggerät)	DM 289,00
☐	Easytizer (Teilsatz wie oben beschrieben)	DM 129,00

Versandkosten: Inland DM 7,50
Ausland DM 10,00
Auslandbestellung **nur** gegen Vorauskasse. Nachnahme zuzgl. DM 3,50 Nachnahmegebühr.

☐ Vorauskasse
☐ Nachnahme

```
156:          beq fett_ein        ; Ja -> verzweigen
157:          cmp.b #$26,d1       ; Fettdruck aus?
158:          beq fett_aus        ; Ja -> verzweigen
159:          cmp.b #$1F,d1       ; HMI setzen?
160:          beq hmi1            ; Ja -> verzweigen
161:          cmp.b #$05,d1       ; VMI setzen?
162:          beq vmi1            ; Ja -> verzweigen
163:          bra exit            ; andere ESC-Sequenzen
                                    kenn ich nicht!!!
164:
165: esc_3byt:
166:          cmp.w #2,esc_flag ; HMI-Sequenz?
167:          bne vmi2            ; Nein -> verzweigen
168:          subq.b #1,d1        ; HMI = N-1
169:          move.b d1,teilung ; Neue Teilung merken
170:          move.b #$80,d4      ; Neue Teilung an SM schicken
171:          bsr senden
172:          move.b teilung,d4
173:          bsr senden
174:          move.w #0,esc_flag ; Flag löschen
175:          bra exit
176:
177: vmi2:    cmp.w #3,esc_flag   ; VMI-Sequenz?
178:          move.w #0,esc_flag ; Flag löschen
179:          bne exit            ; Nein -> dann unbekannt
180:          subq.l #1,d1        ; VMI = N-1
181:          move.b d1,zeile     ; Neue Schrittzahl für LF
                                    merken
182:          bra exit
183:
184: li_rand: move.w #0,spalte   ; Spaltenzähler auf Null
                                    setzen
185:          bra exit
186:
187: hochst: move.w #$f0,d1    ; Vertikale Bewegung um 8/96"
188:          bsr print          ; = 0,5 Grundzeilen zurück
189:          move.w #8 ,d1
190:          bsr print
191:          bra exit
192:
193: tiefst: move.w #$D0,d1    ; Vertikale Bewegung um 8/96"
194:          bsr print          ; = 0,5 Grundzeilen vorwärts
195:          move.w #8 ,d1
196:          bsr print
197:          bra exit
198:
199: lf_back: move.w #$f0,d1   ; Vert.Bewegung u.normalerweise
200:          bsr print          ; 16/96" = 1 Grundzeile zurück
201:          move.b zeile ,d1
202:          bsr print
203:          bra exit
204:
205: unt_ein: bset #1,son_flag  ; Bit 1 für Unterstreichen
                                    setzen
206:          bra exit
207:
208: unt_aus: bclr #1,son_flag  ; Bit 1 für Unterstreichen
                                    löschen
209:          bra exit
210:
211: fett_ein: bset #2,son_flag  ; Bit 2 für Fettschrift
                                    setzen
212:         move.b #$80,d1      ; Horizontalschritt auf
Teilung-1
213:         bsr print           ; Teilung-1 anpassen
214:         move.b teilung,d1
215:         subq.b #1,d1
216:         bsr print
217:         bra exit
218:
219: fett_aus: bclr #2,son_flag  ; Bit 2 für Fettschrift
                                    löschen
220:          move.b #$80,d1     ; Horizontalschritt auf
221:          bsr print          ; eingestellte Teilung
222:          move.b teilung,d1  ; anpassen
223:          bsr print
224:          bra exit
225:
226: hmi1:    move.w #2,esc_flag  ; Flag für HMI-Sequenz
                                    setzen
227:          bra exit
228:
229: vmi1:    move.w #3,esc_flag  ; Flag für VMI-Sequenz
                                    setzen
230:          bra exit
231:
232:                          ; Abfrage nach Sonderzeichen
233: so_zei: cmp.b #$20,d1    ; ASCII-Code = 32
234:         beq space        ; Ja -> Leerschritt
235:         cmp.b #13,d1     ; ASCII-Code = 13
236:         beq cr           ; Ja -> Carriage Return
237:         cmp.b #10,d1     ; ASCII-Code = 10
238:         beq lf           ; Ja -> Linefeed
239:         cmp.b #$1B,d1    ; ASCII-Code = 27
240:         beq esc          ; Ja -> ESC-Zeichen -> Flag
                                                       setzen
241:         cmp.b #8,d1      ; ASCII-Code = 8
242:         beq backsp       ; Ja -> Backspace
243:         cmp.b #12,d1     ; ASCII-Code = 12
244:         beq formfeed     ; JA -> Formfeed
245:         bra exit  ; Andere Sonderzeichen kenn ich nicht
246:
247: esc:    move.w #1,esc_flag  ; Flag für erstes Byte aus
                                   Sequenz setzen
248:         bra exit
249:
250:
251: space:
252:         move.b teilung,d1    ; Leerschritt nicht direkt
                                    ausführen,
253:         and.l #$FF,d1        ; sondern für später
                                    zusammenfassen
254:         add.w d1,step_li     ; und in "step_li" merken
255:         add.w d1,spalte
256:         bra exit
257:
258: backsp: move.w #$84,d1      ; Leerschritt nach links
                                   -> direkt ausführen
259:         bsr print            ; direkt ausführen
260:         move.b teilung,d1
261:         and.w #$ff,d1
262:         sub.w d1,spalte
263:         move.w #0,d1
264:         bsr print
265:         bra exit
266:
267: cr:     move.w spalte,d0     ; Carriage-Return ausführen
                                   dabei auch
268:                              ; noch nicht ausgeführte
                                   Leerschritte
269:         sub.w step_li,d0     ; berücksichtigen
270:         move.w #0,step_li
271:         or.w #$e000,d0
272:         move.w d0,d1
273:         lsr #8,d1
274:         move.w d0,-(sp)
275:         bsr print
276:         move.w (sp)+,d1
277:         and.w #$FF,d1
278:         bsr print
279:         move.w #0,spalte  ; Spaltenzähler zurücksetzen
280:         bra exit
281:
282: lf:     move.w #$D0,d1     ; Linefeed ausführen, Anzahl
283:         bsr print          ; der Schritte steht in #zeil
284:         move.b zeile,d1
285:         bsr print
286:         bra exit
287:
288: formfeed:
289:         move.b #$a3,d1
290:         bsr print
291:         move.b #0,d1
292:         bsr print
293: wait:   bsr lesen
294:         cmp.b #1,d5
295:         bne wait
296:         move.b #$a2,d1
297:         bsr print
298:         move.b #0,d1
299:         bsr print
300:         bra exit
301:
302: exit:
303:         move.w onl_flag,d0
304:         cmp.w #1,d0
305:         bne probl
306:         moveq #-1,d0
307:         bra exit1
308: probl:  moveq #0,d0
309: exit1:  move.l  $4a2,a1
310:         move.l  (a1)+,-(sp)
```

Listing geht weiter...

```
311:         move    (a1)+,-(sp)
312:         move.l  a1,$4a2
313:   movem.l (sp)+,d1-d7/a0-a6
314:         move #$2300,sr
315:         rte
316:
317: normal:
318:         cmp     #8,6(a2)
319:         bne     norml
320:         cmp     #0,8(a2)
321:         bne     norml
322:
323:         moveq   #-1,d0
324:         move.w onl_flag,d2
325:         cmp.w #1,d2
326:         bne probl1
327:         movea.l #$fffa01,a3
328:         move.b (a3),d2
329:         btst #2,d2
330:         beq platz
331:
332:
333: probl1: moveq   #0,d0
334:
335: platz:  rte
336:
337: norml:  move.l  tr_save,a0
338:         jmp     (a0)
339:
340: print:  move.b d1,d4
341:         bsr senden
342:
343:         rts
344:
345:
346:
347: ; ****************************************************
348: em_int:                  ; Interrupt-routine, die von
349:         move #$2700,sr   ; der SM ausgelöst wird
350:         MOVEM.L D0-D6/A0-A3,-(SP)
351:         bsr lesen        ; Empfangenes Zeichen lesen
352:         cmp.b #1,d5      ; Wurde die Online-Taste
                               gedrückt?
353:         bne nionline     ; Nein -> weitere Abfrage
354:         move.w onl_flag,d5 ; Ist die SM schon ONLINE?
355:         cmp.w #1,d5
356:         beq ni_line      ; Ja -> nichts machen
357:         bsr online       ; Nein -> Online schalten
358:         move.w #1,onl_flag ; Flag setzen
359:         bra fertig
360: nionline:
361:         cmp.b #2,d5 ; Wurde die OFFLINE-Taste gedrückt?
362:         bne ni_line ; Nein -> nichts machen
363:         move.w onl_flag,d5 ; Ist SM noch OFFLINE?
364:         cmp.w #0,d5
365:         beq ni_line  ; Ja -> nichts machen
366:         bsr offline  ; Nein -> Offline schalten
367:         bsr offline
368:         move.w #0,onl_flag ; Flag zurücksetzen
369:         bra fertig
370: ni_line:
371:         move.b #0,d4 ; zwei "dummy-zeichen" senden
372:         bsr senden
373:         move.b #0,d4
374:         bsr delay2
375:         bsr senden
376:         bsr delay2
377:
378: fertig:
379:         move.l #$FFFA0F,a0  ; Interrupt-Service-Bit
                                  löschen
380:         bclr #4,(a0)
381:         movem.l (sp)+,d0-d6/a0-a3
382:         move #$2300,sr
383:         rte
384:
385: online:
386:         MOVEM.L D0-D7/A0-A4,-(SP)
387:         bsr raus
388:         bsr delay2
389:         lea onl_tab,a1
390:         move.l #tab_end-onl_tab-1,d0
391: loop9:
392:         move.b (a1)+,d4
393:         bsr senden
394:         bsr delay1
395:         move.b #0,d4
396:         bsr senden
397:         bsr delay2
398:         bsr raus
399:         dbra d0,loop9
400:         movem.l (sp)+,d0-d7/a0-a4
401:         bsr delay2
402:         move.b #$82,d4
403:         bsr senden
404:         bsr delay1
405:         move.b #$1F,d4
406:         bsr senden
407:         bsr delay2
408:         move.b #$80,d4
409:         bsr senden
410:         bsr delay1
411:         move.b teilung,d4
412:         bsr senden
413:         bsr delay2
414:         move.w #0,spalte
415:         move.w #0,esc_flag
416:         move.w #0,son_flag
417:         rts
418:
419: **************************************************
420: offline:
421:         lea offl_tab,a1
422:         move.w #tab_end2-offl_tab-1,d0
423:         move.w #0,d6
424:
425: loop1:  move.b (a1)+,d4
426:         bsr senden
427:         bsr delay2
428:         dbra d0,loop1
429:         rts
430: **************************************************
431: senden:
432:         movea.l #$fffa01,a3
433:         move.b (a3),d7
434:         btst #2,d7
435:         bne senden
436: send1:
437:         movea.l #$fffa2f,a3
438:         move.b d4,(a3)
439: send2:
440:         movea.l #$fffa01,a3
441:         move.b (a3),d7
442:         btst #2,d7
443:         beq send2
444:         rts
445:
446: lesen:
447:         move.w #100,d6
448: les1:   sub #1,d6
449:         cmp.w #0,d6
450:         beq raus
451:         movea.l #$fffa2b,a3
452:         move.b (a3),d5
453:         and.w #$80,d5
454:         cmp.b #0,d5
455:         beq les1
456: raus:
457:         movea.l #$FFFa2F,a3
458:         move.b (a3),d5
459:         and.l  #$FF,d5
460:         rts
461:
462: delay1:
463:         move.l d0,-(sp)
464:         move.l #$3fff,d0
465: loop2:
466:         dbra d0,loop2
467:         move.l (sp)+,d0
468:         rts
469: delay2:
470:         move.l d0,-(sp)
471:         move.l #$ffff,d0
472: loop3:
473:         move.l d0,d0
474:         move.l d0,d0
475:         dbra d0,loop3
476:         move.l (sp)+,d0
477:         rts
478:
```

Listing geht weiter...

```
479:  init:
480:            move.w #-1,-(sp)
481:            move.w #-1,-(sp)
482:            move.w #-1,-(sp)
483:            move.w #-1,-(sp)
484:            move.w #2,-(sp)
485:            move.w #2,-(sp)
486:            move.w #15,-(sp)
487:            trap #14
488:            add.l #14,sp
489:
490:            move.w #1,-(sp)
491:            move.w #26,-(sp)
492:            trap #14
493:            addq.l #4,sp
494:
495:            move.w #10,-(sp)
496:            move.w #26,-(sp)
497:            trap #14
498:            addq.l #4,sp
499:
500:            move.w #9,-(sp)
501:            move.w #26,-(sp)
502:            trap #14
503:            addq.l #4,sp
504:
505:            move.w #11,-(sp)
506:            move.w #26,-(sp)
507:            trap #14
508:            addq.l #4,sp
509:            rts
510:
511:            move.w #2,-(sp)
512:            move.w #26,-(sp)
513:            trap #14
514:            addq.l #4,sp
515:
516:
517:     .data
518:  onl_tab:
519:            dc.b $A1,$A4,$A2,0
520:  tab_end:
521:            move.w d0,d0
522:  offl_tab:
523:            dc.b $A3,0,$A0,0
524:  tab_end2:
525:            move.w d0,d0
526:
527:  onl_flag: dc.w 0
528:
529:  zei_satz:
530:     dc.b   $3E,$92         ;       !
531:     dc.b   $2B,$90         ;       "
532:     dc.b   $4C,$99         ;       #
533:     dc.b   $27,$97         ;       $
534:     dc.b   $43,$97         ;       %
535:     dc.b   $22,$99         ;       &
536:     dc.b   $0E,$8C         ;       '
537:     dc.b   $3A,$92         ;       (
538:     dc.b   $31,$92         ;       )
539:     dc.b   $29,$92         ;       *
540:     dc.b   $0F,$90         ;       +
541:     dc.b   $02,$8E         ;       ,
542:     dc.b   $03,$8E         ;       -
543:     dc.b   $01,$8C         ;       .
544:     dc.b   $3C,$92         ;       /
545:     dc.b   $19,$94         ;       0
546:     dc.b   $10,$94         ;       1
547:     dc.b   $11,$94         ;       2
548:     dc.b   $12,$94         ;       3
549:     dc.b   $13,$94         ;       4
550:     dc.b   $14,$94         ;       5
551:     dc.b   $15,$94         ;       6
552:     dc.b   $16,$92         ;       7
553:     dc.b   $17,$97         ;       8
554:     dc.b   $18,$94         ;       9
555:     dc.b   $36,$90         ;       :
556:     dc.b   $35,$90         ;       ;
557:     dc.b   $4A,$92         ;       <
558:     dc.b   $34,$92         ;       =
559:     dc.b   $0D,$92         ;       >
560:     dc.b   $2D,$94         ;       ?
561:     dc.b   $40,$99         ;     §
562:     dc.b   $24,$99         ;       A
563:     dc.b   $1C,$99         ;       B
564:     dc.b   $2A,$97         ;       C
565:     dc.b   $2C,$99         ;       D
566:     dc.b   $1A,$97         ;       E
567:     dc.b   $1D,$94         ;       F
568:     dc.b   $45,$99         ;       G
569:     dc.b   $39,$97         ;       H
570:     dc.b   $2F,$92         ;       I
571:     dc.b   $42,$94         ;       J
572:     dc.b   $3B,$97         ;       K
573:     dc.b   $26,$94         ;       L
574:     dc.b   $37,$99         ;       M
575:     dc.b   $2E,$99         ;       N
576:     dc.b   $3D,$99         ;       O
577:     dc.b   $1E,$99         ;       P
578:     dc.b   $41,$99         ;       Q
579:     dc.b   $28,$99         ;       R
580:     dc.b   $1F,$97         ;       S
581:     dc.b   $25,$94         ;       T
582:     dc.b   $30,$99         ;       U
583:     dc.b   $21,$94         ;       V
584:     dc.b   $32,$99         ;       W
585:     dc.b   $3F,$94         ;       X
586:     dc.b   $23,$94         ;       Y
587:     dc.b   $20,$97         ;       Z
588:     dc.b   $3A,$92         ;       ( für [
589:     dc.b   $29,$92         ;       ; für \
590:     dc.b   $31,$92         ;       ) für ]
591:     dc.b   $0C,$8C         ;       ^
592:     dc.b   $33,$90         ;       _
593:     dc.b   $48,$8C         ;       `
594:     dc.b   $5E,$94         ;       a
595:     dc.b   $62,$97         ;       b
596:     dc.b   $58,$92         ;       c
597:     dc.b   $60,$97         ;       d
598:     dc.b   $5D,$94         ;       e
599:     dc.b   $0B,$92         ;       f
600:     dc.b   $59,$97         ;       g
601:     dc.b   $57,$94         ;       h
602:     dc.b   $5F,$92         ;       i
603:     dc.b   $07,$92         ;       j
604:     dc.b   $55,$94         ;       k
605:     dc.b   $05,$92         ;       l
606:     dc.b   $06,$97         ;       m
607:     dc.b   $5A,$94         ;       n
608:     dc.b   $63,$94         ;       o
609:     dc.b   $56,$97         ;       p
610:     dc.b   $4F,$97         ;       q
611:     dc.b   $5B,$92         ;       r
612:     dc.b   $5C,$94         ;       s
613:     dc.b   $4D,$92         ;       t
614:     dc.b   $61,$94         ;       u
615:     dc.b   $04,$92         ;       v
616:     dc.b   $08,$94         ;       w
617:     dc.b   $4E,$92         ;       x
618:     dc.b   $54,$94         ;       y
619:     dc.b   $64,$92         ;       z
620:     dc.b   $3A,$92         ;       (
621:     dc.b   $2F,$92         ;       I
622:     dc.b   $31,$92         ;       )
623:     dc.b   $83,0           ;       nicht auf TR vorhanden
624:     dc.b   $83,0           ;       nicht auf TR vorhanden
625:     dc.b   $83,0           ;       nicht auf TR vorhanden
626:     dc.b   $51,$97         ;       ü
627:     dc.b   $83,0           ;       nicht auf TR vorhanden
628:     dc.b   $83,0           ;       nicht auf TR vorhanden
629:     dc.b   $53,$97         ;       ä
630:     dc.b   $83,0           ;       nicht auf TR vorhanden
631:     dc.b   $83,0           ;       nicht auf TR vorhanden
632:     dc.b   $83,0           ;       nicht auf TR vorhanden
633:     dc.b   $83,0           ;       nicht auf TR vorhanden
634:     dc.b   $83,0           ;       nicht auf TR vorhanden
635:     dc.b   $83,0           ;       nicht auf TR vorhanden
636:     dc.b   $83,0           ;       nicht auf TR vorhanden
637:     dc.b   $83,0           ;       nicht auf TR vorhanden
638:     dc.b   $83,0           ;       nicht auf TR vorhanden
639:     dc.b   $4b,$99         ;       Ä
640:     dc.b   $83,0           ;       nicht auf TR vorhanden
641:     dc.b   $83,0           ;       nicht auf TR vorhanden
642:     dc.b   $83,0           ;       nicht auf TR vorhanden
643:     dc.b   $83,0           ;       nicht auf TR vorhanden
644:     dc.b   $83,0           ;       nicht auf TR vorhanden
645:     dc.b   $52,$97         ;       ö
646:     dc.b   $83,0           ;       nicht auf TR vorhanden
647:     dc.b   $83,0           ;       nicht auf TR vorhanden
648:     dc.b   $83,0           ;       nicht auf TR vorhanden
```

Listing geht weiter...

```
649:    dc.b  $83,0      ;     nicht auf TR vorhanden
650:    dc.b  $49,$99    ;     Ö
651:    dc.b  $47,$99    ;     Ü
652:    dc.b  $83,0      ;     nicht auf TR vorhanden
653:    dc.b  $1B,$99    ;     pfund
654:    dc.b  $83,0      ;     nicht auf TR vorhanden
655:    dc.b  $50,$97
656:    dc.b  $83,0      ;     nicht auf TR vorhanden
657:    dc.b  $83,0      ;     nicht auf TR vorhanden
658:    dc.b  $83,0      ;     nicht auf TR vorhanden
659:    dc.b  $83,0      ;     nicht auf TR vorhanden
660:    dc.b  $83,0      ;     nicht auf TR vorhanden
661:    dc.b  $83,0      ;     nicht auf TR vorhanden
662:    dc.b  $83,0      ;     nicht auf TR vorhanden
663:    dc.b  $83,0      ;     nicht auf TR vorhanden
664:    dc.b  $83,0      ;     nicht auf TR vorhanden
665:    dc.b  $83,0      ;     nicht auf TR vorhanden
666:    dc.b  $83,0      ;     nicht auf TR vorhanden
667:    dc.b  $83,0      ;     nicht auf TR vorhanden
668:    dc.b  $83,0      ;     nicht auf TR vorhanden
669:    dc.b  $83,0      ;     nicht auf TR vorhanden
670:    dc.b  $83,0      ;     nicht auf TR vorhanden
671:    dc.b  $83,0      ;     nicht auf TR vorhanden
672:    dc.b  $83,0      ;     nicht auf TR vorhanden
673:    dc.b  $83,0      ;     nicht auf TR vorhanden
674:    dc.b  $83,0      ;     nicht auf TR vorhanden
675:    dc.b  $83,0      ;     nicht auf TR vorhanden
676:    dc.b  $83,0      ;     nicht auf TR vorhanden
677:    dc.b  $83,0      ;     nicht auf TR vorhanden
678:    dc.b  $83,0      ;     nicht auf TR vorhanden
679:    dc.b  $83,0      ;     nicht auf TR vorhanden
680:    dc.b  $83,0      ;     nicht auf TR vorhanden
681:    dc.b  $83,0      ;     nicht auf TR vorhanden
682:    dc.b  $83,0      ;     nicht auf TR vorhanden
683:    dc.b  $83,0      ;     nicht auf TR vorhanden
684:    dc.b  $83,0      ;     nicht auf TR vorhanden
685:    dc.b  $83,0      ;     nicht auf TR vorhanden
686:    dc.b  $83,0      ;     nicht auf TR vorhanden
687:    dc.b  $83,0      ;     nicht auf TR vorhanden
688:    dc.b  $83,0      ;     nicht auf TR vorhanden
689:    dc.b  $83,0      ;     nicht auf TR vorhanden
690:    dc.b  $83,0      ;     nicht auf TR vorhanden
691:    dc.b  $83,0      ;     nicht auf TR vorhanden
692:    dc.b  $83,0      ;     nicht auf TR vorhanden
693:    dc.b  $83,0      ;     nicht auf TR vorhanden
694:    dc.b  $83,0      ;     nicht auf TR vorhanden
695:    dc.b  $83,0      ;     nicht auf TR vorhanden
696:    dc.b  $83,0      ;     nicht auf TR vorhanden
697:    dc.b  $83,0      ;     nicht auf TR vorhanden
698:    dc.b  $83,0      ;     nicht auf TR vorhanden
699:    dc.b  $83,0      ;     nicht auf TR vorhanden
700:    dc.b  $83,0      ;     nicht auf TR vorhanden
701:    dc.b  $83,0      ;     nicht auf TR vorhanden
702:    dc.b  $83,0      ;     nicht auf TR vorhanden
703:    dc.b  $83,0      ;     nicht auf TR vorhanden
704:    dc.b  $83,0      ;     nicht auf TR vorhanden
705:    dc.b  $83,0      ;     nicht auf TR vorhanden
706:    dc.b  $83,0      ;     nicht auf TR vorhanden
707:    dc.b  $83,0      ;     nicht auf TR vorhanden
708:    dc.b  $83,0      ;     nicht auf TR vorhanden
709:    dc.b  $83,0      ;     nicht auf TR vorhanden
710:    dc.b  $83,0      ;     nicht auf TR vorhanden
711:    dc.b  $83,0      ;     nicht auf TR vorhanden
712:    dc.b  $83,0      ;     nicht auf TR vorhanden
713:    dc.b  $83,0      ;     nicht auf TR vorhanden
714:    dc.b  $83,0      ;     nicht auf TR vorhanden
715:    dc.b  $83,0      ;     nicht auf TR vorhanden
716:    dc.b  $83,0      ;     nicht auf TR vorhanden
717:    dc.b  $83,0      ;     nicht auf TR vorhanden
718:    dc.b  $50,$97    ;     P
720:    dc.b  $83,0      ;     nicht auf TR vorhanden
721:    dc.b  $83,0      ;     nicht auf TR vorhanden
722:    dc.b  $83,0      ;     nicht auf TR vorhanden
723:    dc.b  $83,0      ;     nicht auf TR vorhanden
724:    dc.b  $83,0      ;     nicht auf TR vorhanden
725:    dc.b  $83,0      ;     nicht auf TR vorhanden
726:    dc.b  $83,0      ;     nicht auf TR vorhanden
727:    dc.b  $83,0      ;     nicht auf TR vorhanden
728:    dc.b  $83,0      ;     nicht auf TR vorhanden
729:    dc.b  $83,0      ;     nicht auf TR vorhanden
730:    dc.b  $83,0      ;     nicht auf TR vorhanden
731:    dc.b  $83,0      ;     nicht auf TR vorhanden
732:    dc.b  $83,0      ;     nicht auf TR vorhanden
733:    dc.b  $83,0      ;     nicht auf TR vorhanden
734:    dc.b  $83,0      ;     nicht auf TR vorhanden
735:    dc.b  $83,0      ;     nicht auf TR vorhanden
736:    dc.b  $83,0      ;     nicht auf TR vorhanden
737:    dc.b  $83,0      ;     nicht auf TR vorhanden
738:    dc.b  $83,0      ;     nicht auf TR vorhanden
739:    dc.b  $83,0      ;     nicht auf TR vorhanden
740:    dc.b  $83,0      ;     nicht auf TR vorhanden
741:    dc.b  $83,0      ;     nicht auf TR vorhanden
742:    dc.b  $83,0      ;     nicht auf TR vorhanden
743:    dc.b  $83,0      ;     nicht auf TR vorhanden
744:    dc.b  $83,0      ;     nicht auf TR vorhanden
745:    dc.b  $83,0      ;     nicht auf TR vorhanden
746:    dc.b  $83,0      ;     nicht auf TR vorhanden
747:    dc.b  $83,0      ;     nicht auf TR vorhanden
748:    dc.b  $83,0      ;     nicht auf TR vorhanden
749:    dc.b  $83,0      ;     nicht auf TR vorhanden
750:    dc.b  $83,0      ;     nicht auf TR vorhanden
751:    dc.b  $83,0      ;     nicht auf TR vorhanden
752:    dc.b  $83,0      ;     nicht auf TR vorhanden
753:
754:
755:  buffer=  0
756:  len=     4
757:  head=    6
758:  tail=    8
759:
760:          .bss
761:  step_li:    ds.w 1
762:  step_re:    ds.w 1
763:  spalte:     ds.w 1
764:  esc_flag:   ds.w 1
765:  son_flag:   ds.w 1
766:  tr_save:    ds.l 1
767:  zeile:      ds.b 1
768:  teilung:    ds.b 1
769:
770:    end
```

Listing : Der IFD-Emulators für TA Gabriele 9009 in Assembler

ENDE

Flexible Modulprogrammierung mit ADIMENS Talk

Teil 1

Menüsteuerung

Zweifellos hat das Datenbanksystem ADIMENS eine breite Akzeptanz bei den ATARI Anwendern gefunden. Einen wesentlichen Einfluß hat dabei die integrierte Kommandosprache ADIMENS Talk, die über die Befehle zur reinen Datenmanipulation hinausgeht und eine komplette Anwendungsprogrammierung zuläßt.

Willkommen zum 1. Teil der flexiblen Modulprogrammierung mit ADIMENS Talk. ADIMENS ist meines Erachtens eines der benutzerfreundlichsten und leistungsfähigsten Datenbanksysteme. Falls Sie es bereits einsetzen, vertieft diese Serie Ihre Kenntnisse. Wenn Sie es zum erstenmal benutzen, werden Sie feststellen, wie angenehm das Arbeiten mit ihm im allgemeinen und mit ADIMENS Talk im besonderen ist.

Der Inhalt dieser Serie ist auf die große Gruppe der semiprofessionellen Anwender abgestimmt, welcher mit diesen flexiblen Programmmoduln ein Instrument zur schnellen Entwicklung von EDV-Projekten kennenlernen wird. Neben den maschinenorientierten (z.B. Assembler) oder den problemorientierten Sprachen (z.B. Cobol), welche einen großen Aufwand für die Entwicklung von Programmen erfordern, hat man mit der Abfragesprache ADIMENS Talk eine integrierte Kommandosprache in der Hand, die über die Befehle zur reinen Datenmanipulation hinausgeht und eine komplette Anwendungsprogrammierung zuläßt.

Allgemeine Gesichtspunkte

ADIMENS Talk wird als Programmiersprache in einer Umgebung verwendet, in der die Benutzer keine professionellen Programmierer sind. Ich meine, daß Menschen, die Datenbanksysteme wie ADIMENS verwenden, nicht zwangsläufig zu Profi-Programmierern werden sollten. Andererseits sollte jeder zu den leistungsfähigen Computern und einer so leistungsfähigen Sprache, wie sie ADIMENS Talk bietet, Zugang haben.

Meiner Vorstellung nach sind Computer und Anwendungsprogramme schlicht Werkzeuge, die man für eine bestimmte Aufgabe benutzt und dann wieder beiseite legt. Sie sollten nicht mehr Zeit in Anspruch nehmen, als die zu erledigende Aufgabe erfordert.

Insbesondere die letztgenannte Eigenschaft ist für den Benutzer von größter Bedeutung, der auf die konventionellen Sprachen wie C, Fortran oder Pascal völlig verzichten will und eine Sprache auf höherem Niveau benutzen will, die jenen an Mächtigkeit weit überlegen ist.

Modulkonzeption

Mehrere Programmteile, die in einem logischen Zusammengang stehen, können zu Moduln zusammengefaßt werden. In den in dieser Serie vorgestellten flexiblen Moduln mit ADIMENS Talk wird jede Teilaufgabe, die in den meisten Anwendungsprogrammen bei Datenbankoperationen auftaucht, systematisch vorgestellt.

Dabei war stets das Ziel, Moduln zu erstellen, so daß eine Software in Form von austauschbaren Bauteilen zusammengestellt werden kann.

Ein wichtiger Bestandteil der Programmentwicklung wird dabei die Möglichkeit sein, die Datenbank-Abfragesprache zu erlernen und dann später diese Module nach eigenen Wünschen und Vorstellungen in ein Anwendungsprogramm zu implementieren.

Im Voraus darf hier schon gesagt werden, daß folgende Modultypen erarbeitet wurden.

- **Startmodul,** welches die Zugriffsrecht erteilt und datenmäßig er faßt.
- **Modul für die Menütechnik,** zum Auswählen mehrerer Funktionen und zum Aufruf der entsprechenden Unterprogramme.
- **Modul zum Erfassen von Merkmalen** verschiedener Dateien und zum Abspeichern neu erstellter Datensätze.
- **Modul zum Suchen von Daten.** Dabei kann nach zwei Merkmalen gesucht werden, die numerischer als auch alphanumerischer Form sein können.Ebenfalls sind fragmentarische Eingaben erlaubt und eine Suche nach Merkmalen in einem bestimmten Zahlenbereich.

- Die gefundenen Datensätze können angezeigt oder speziell gewünschte Datensätze ausgedruckt, sortiert und beliebig aufbereitet werden.
- **Modul zum Ändern von Daten.** Die erfaßten Datensätze können gelöscht oder verändert werden.

Alle Funktionen werden anhand praxisnaher Beispiele vorgeführt und können vom Leser leicht nachvollzogen werden. Um den systematischen Aufbau der Moduln nicht zu beeinträchtigen, wird bewußt auf das Besprechen sämtlicher Kommandos und aller ihrer Varianten verzichtet, deren Aufzählung mehr verwirren würde als klären.

Das bedeutet für den Leser ganz konkret, daß er ADIMENS Talk schon nach kurzer Zeit für einfache Anwendungen nutzbar und das weitere Vordringen in verfeinerte Datenbanktechniken von seinen individuellen Bedürfnissen abhängig machen kann.

Die Programmierphilosophie

So selbsterklärend wie die Programmierschnittstelle von ADIMENS zu ADIMENS Talk ist wohl keine andere zur Zeit auf dem Markt befindliche Sprache. Weil aber Menschen unglaublich faul und uneffizient sein können, kann man in dieser Computersprache natürlich auch Fehler und Unsinn programmieren. Die Anziehungskraft von ADIMENS Talk besteht in den strukturierten Programmtexten und den leicht lesbaren Kommentaren.

Die Möglichkeit, ziemlich schnell Programme schreiben oder Probleme lösen zu können, sollte aber nicht daran hindern, die Programmtexte klar strukturiert und kommentiert zu erstellen.

Die Entwicklung eines Anwendungsprogrammes ist eine kreative Tätigkeit. Sie ist z.B. mit einer Konstruktionstätigkeit im ingenieurwissenschaftlichen Bereich zu vergleichen. Die ingenieurmäßige Entwicklung der Software, unter dem Namen **Software-Engineering** bekannt, wendet Methoden, Hilfmittel und Konzepte an, um festgelegte technische, ökonomische und soziale Ziele zu erreichen. Die Entwicklung von Software geschieht nach **Abb. 1** in er Analysephase, Spezifikationsphase, Entwurfsphase, Programmierphase, Implementierungsphase, Testphase und Wartungsphase. Auf die einzelnen Phasen soll hier nicht näher eingegangen werden. Interessierte Leser

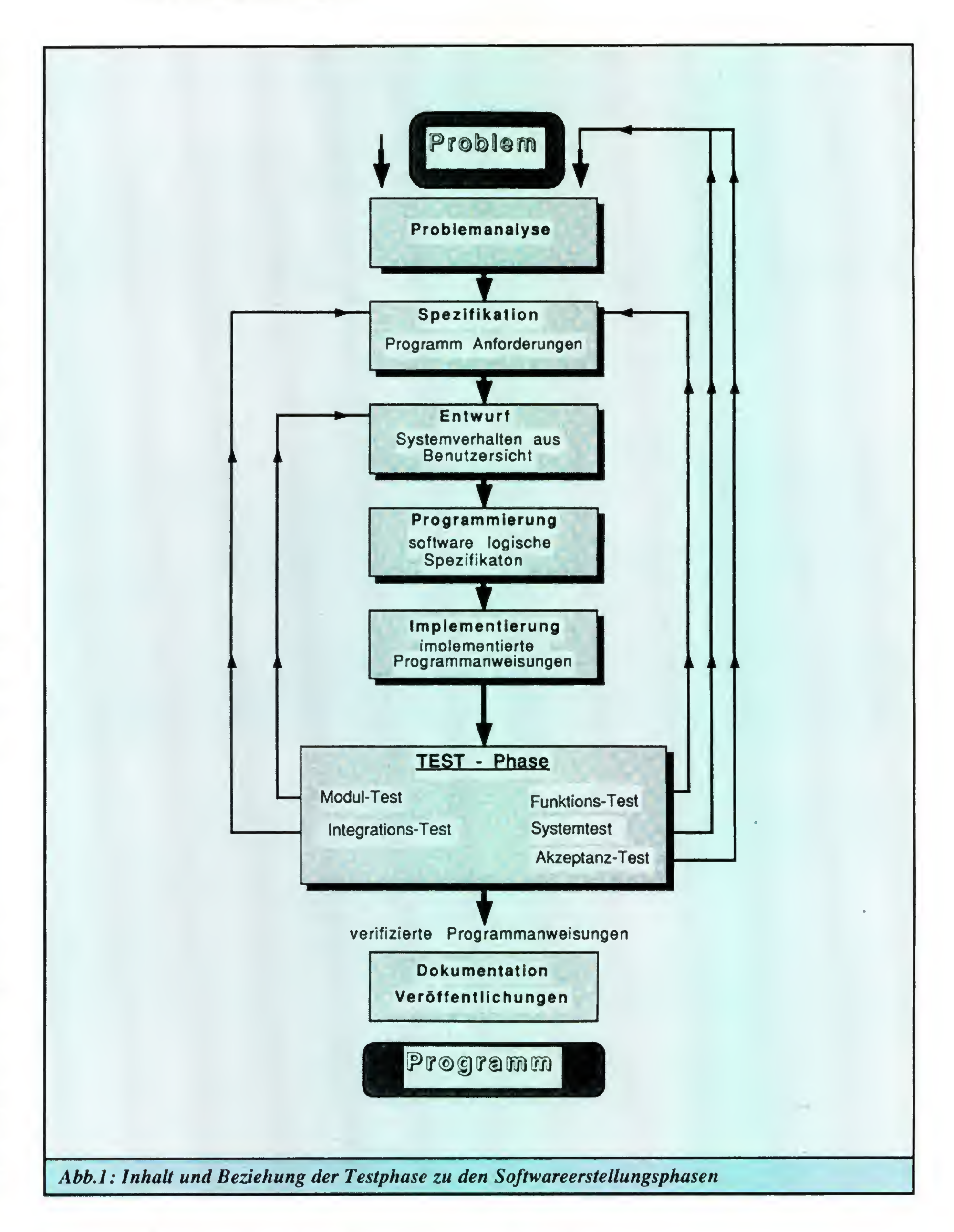


Abb.1: Inhalt und Beziehung der Testphase zu den Softwareerstellungsphasen

können jedoch eine sehr gute Beschreibung dieser einzelnen Phasen in /1/ nachlesen.

Programmiermethoden

Komplizierte und komplexe Probleme werden meiner Meinung nach am besten dadurch gelöst, daß sie **hierarchisch strukturiert** werden und dann vom **Kopf (TOP)** aus gelöst werden. Den Anfang bildet eine globale Aufgabenstellung, die in mehrere Hauptfunktionen und weitere Unterfunktionen unterteilt wird. Das Ergebnis einer derartigen Aufteilung sind klare und übersichtliche Programmteile. Auf jeder Ebene wird das Problem vollständig dargestellt, da jedes Programm auf der nächst höheren Ebene durch seinen Aufruf repräsentiert wird. Ein Vergleich der Darstellung eines Problems auf verschiedenen Hierarchieebenen führt dazu, daß die höhere Ebene angibt, was zu tun ist, während die tiefere angibt, wie es zu tun ist.

Anders ausgedrückt : Es steht bei der hierarchischen Gliederung die Nummernstelle oder der Nummernteil des Moduls in eindeutiger Abhängigkeit von der vorhergehenden Stelle (siehe **Abb. 2**). Ein typisches Beispiel für eine solche hierarchische Gliederung ist das Postleitzahlensystem. Eine Gruppe wird nur dann eindeutig gekennzeichnet, wenn alle vorhergehenden Stellen mit angeführt werden.

Diese ebene Entwurfsstrategie wird auch als " **TOP-DOWN-Entwurf** " bezeichnet, wobei die Vorgehensweise, nämlich die wiederholte Verfeinerung eines allgemeinen Lösungsansatzes - " **Stepwise Refinement** " - beschrieben wird. Dabei ist der TOP-DOWN-Entwurf kein einmaliger Vorgang. Er wiederholt sich quasi auf jeder Ebene, d.h. " Untermodule " werden nach dem gleichen Muster in weitere Untermodule zerlegt usw.

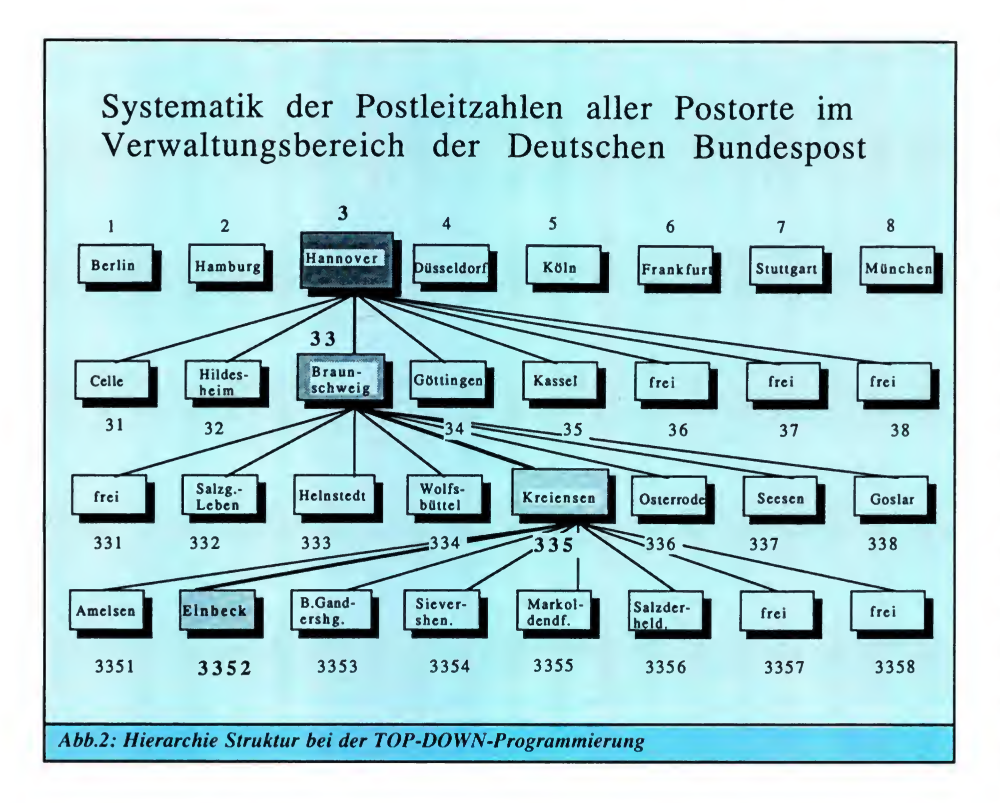


Abb.2: Hierarchie Struktur bei der TOP-DOWN-Programmierung

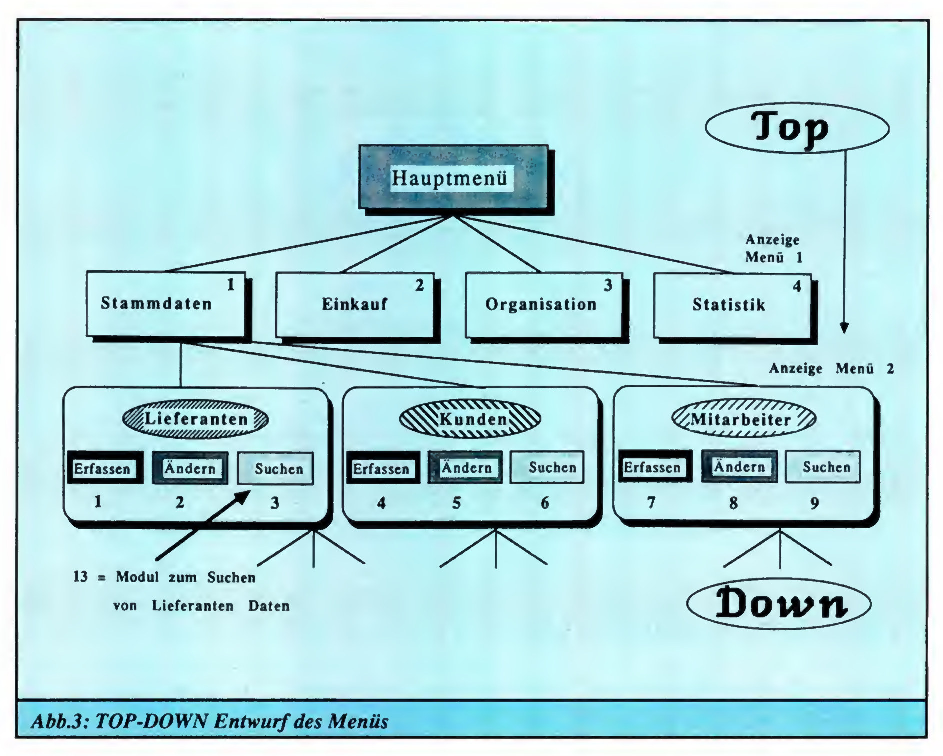


Abb.3: TOP-DOWN Entwurf des Menüs

Konzeption der Menüsteuerung

Beziehen wir dies auf unser Anwendungsprogramm, so müssen wir uns zuerst einmal überlegen, wie unsere Menü-Anzeige aussehen soll. Aus **Abb.3** wird ersichtlich, daß dem Anwender laut Anzeige des 1. Menüs 4 Möglichkeiten zur Auswahl geboten werden. Wählt man die " 1 " für Stammdaten, hat man 9 Auswahlmöglichkeiten. Die Aufbaustruktur ist also hierarchisch gegliedert und entspricht dem TOP-DOWN-Entwurf.

Man sollte sich am Anfang eines Aufbaus zuallererst einmal Gedanken darüber machen, welche Teilaufgaben implementiert werden sollen. Danach sollte die Struktur so geordnet werden, daß man einen systematischen und klar definierter Aufbau vorliegen hat. Sie sollten dabei immer beachten, daß sich jedes Untermodul in eindeutiger Abhängigkeit von dem vorhergehenden Modul befindet.

Es gibt einen alten Begriff in der Computerwelt : " benutzerfreundlich ". Unter **Benutzerfreundlichkeit** eines Anwendungsprogrammes wird die Zufriedenheit eines Benutzers bei der Handhabung seines Programmes verstanden. Die Software-Ergonomie spielt dabei eine wesentliche Rolle. **Software-Ergonomie** ist die nach arbeitspsychologischen Erfordernissen durchgeführte Gestaltung der Software.

Zusammenfassend kann man sagen, daß nur klare Strukturen, eindeutig ansprechbare Menüpunkte und eine extrem kurze Informationseingabe Sie ein Minimum an Zeitaufwand und Nerven kosten.

Nachdem Sie ihre Programmiermethoden sorgfältig entworfen haben, kann mit der Programmierung begonnen werden. Als Anhaltspunkt kann man sagen, daß die Programmierphase in etwa die gleiche Zeit benötigt wie die Vorbereitung und Organisation. Die letzte anschließende Phase an die Programmierung ist die Testphase. In der Regel werden sich in dieser Phase noch Programmänderungswünsche ergeben. Die Testphase wird in etwa so lange wie die Programmierungsphase oder die Vorbereitungsphase dauern.

Das Struktogramm

Ein Programm besteht im Prinzip nur aus einer Reihe von **fortlaufenden Befehlen,** die der Reihe nach abgearbeitet werden. In einem Struktogramm sieht das folgendermaßen aus (siehe **Abb. 4**): Jeder Kasten stellt einen Befehl bzw. eine **Anweisungsgruppe** dar. Die Anweisungsgruppen sind am linken oberen Ende mit einem numerierten Kästchen gekennzeichnet. Das hat den besonderen Vorteil, daß im späteren Quelltext des Programmes diese Anweisungsgruppe in einer Kommentarzeile ebenfalls gekennzeichnet werden kann. Damit findet man sich wesentlich schneller in dem Programm zurecht, oder man kann bei einem längeren Programm direkt nach dieser Zeichenfolge suchen lassen. Die dargestellte Anweisungskennzeichnung im Quelltext in **Abb. 4 unten** zeigt die Struktur des Programmes.

Wichtig ist, daß in dieser Phase der Arbeit versucht werden sollte, keine ADIMENS Talk-Syntax zu verwenden sondern Klartext. Zum Beispiel " Eingabe Identnummer " statt " @ 22,14 SAY " Bitte geben Sie die Identnummer ein " GET PICTURE " ###### ". Das ist äußerst praktisch, da man dann ein Struktogramm ohne Probleme in verschiedene Hochsprachen umsetzen kann. Ebenso werden die Übersichtlichkeit auf der einen und die Verständlichkeit und die Modifikationsfähigkeit des Programmes auf der

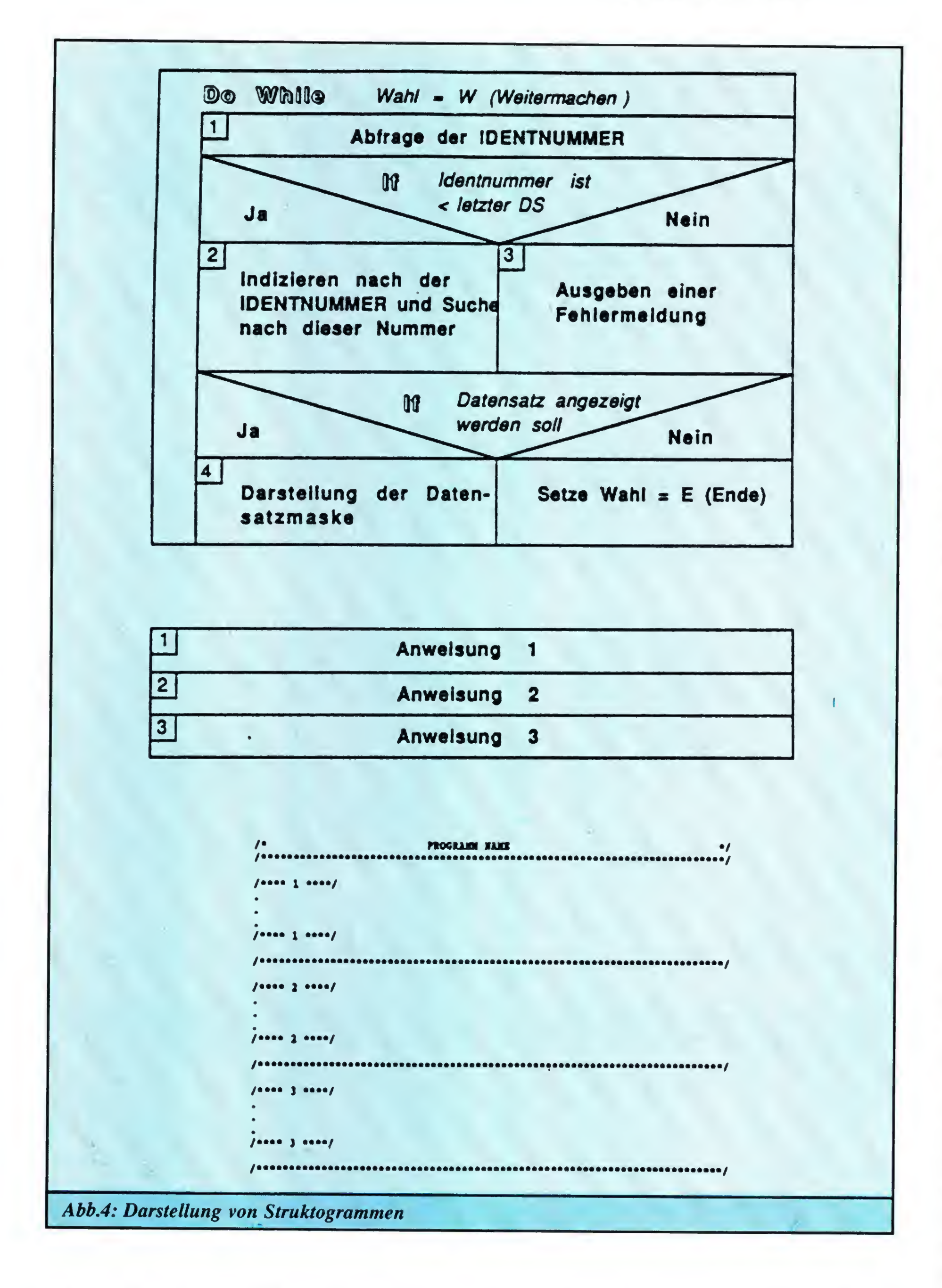


Abb.4: Darstellung von Struktogrammen

anderen Seite wesentlich verbessert.

Die DO WHILE-Schleifen-Struktur

Nun kommt man gleich zu der ersten Form des Struktogrammes - der DO WHILE-Schleife. Sie prüft bereits am Anfang, ob die Bedingung erfüllt ist und läuft nur dann in den Programmblock innerhalb der Schleife hinein. Ist die Bedingung nicht erfüllt, wird das Programm hinter dem Blockende fortgesetzt. Die Schleifen - Konstruktion in ADIMENS lautet:

```
DO WHILE  < Bedingung >
      < Anweisungsblock >
ENDDO
```

Mit WHILE wird ein Anweisungsblock SO LANGE erneut abgearbeitet, bis die einleitende Bedingung nicht mehr erfüllt ist. Als Bedingung kann jeder logische Ausdruck angegeben werden.

Die Überprüfung dieser Bedingung erfolgt nur einmal pro Durchgang, und zwar am Anfang der Schleife (—> also auch vor dem ersten Durchgang). Es gibt damit die Möglichkeit, daß eine DO WHILE Schleife bei einer entsprechenden Bedingung erst gar nicht zum Laufen kommt.

Beispiel :

```
i = 0
DO WHILE ( i < 5 )
   i = i + 1
   ?? i
ENDDO
```

Bildschirmausgabe : 1 2 3 4 5

Die Darstellung im Struktogramm und die konzeptionelle Umsetzung in den Quelltext bei ADIMENS Talk sieht man in *Abb. 5*. Hier ist noch anzumerken, daß die Befehle **LOOP** und **EXIT** bei der Programmierung in dieser Serie **nicht verwendet** werden. Ein Sprung an den Anfang oder ein Abbruch einer Schleife wird deshalb ignoriert.

Die IF-THEN-ELSE-Abfrage

Ein anderer Bestandteil der strukturierten Programmierung ist die "IF- THEN-ELSE" Abfrage. Versucht man diesen Befehl zu übersetzen, lautet er " WENN-DANN-ANSONSTEN ". Zur Steuerung größerer Programmabläufe ist diese Kontrollstruktur unersetzlich. Man wird bald erkennen, daß die wesentlichen Programmstrukturen mit diesem Befehl erstellt werden können.

Die Anweisung lautet in ADIMENS:

```
IF    < Bedingung >
      Anweisungsblock 1
ELSE
      Anweisungsblock 2
ENDIF
```

Die Anweisungen nach ELSE werden dann ausgeführt, wenn die Bedingung der IF-Anweisung nicht erfüllt ist. Sollen mehrere Alternativen abgearbeitet werden, kann eine reine Verwendung der IF-Anweisungen sehr rasch unübersichtlich werden. Vor allem muß jeder mit der IF < Bedingung > neu begonnene Block auch wieder an irgend einer Stelle des Programmes mit ENDIF abgeschlossen werden.

Im Struktogramm (siehe **Abb. 5**) wird dies noch deutlicher. Benötigt man den ELSE-Block nicht, so läßt man ihn einfach weg, und der Platz innerhalb der Anweisung 2 bleibt leer.

Eine einfachere und lesbare Struktur für die Auswahl eines bestimmten Falles ist durch eine CASE-Schachtelung möglich:

```
DO CASE
   CASE < Bedingung 1 >
         < Anweisungsblock 1 >
   CASE < Bedingung 2 >
         < Anweisungsblock 2 >
   ..........
   CASE < Bedingung n >
         <Anweisungsblock n>
ENDCASE
```

Es wird höchstens ein Anweisungsblock abgearbeitet, nämlich genau derjenige,

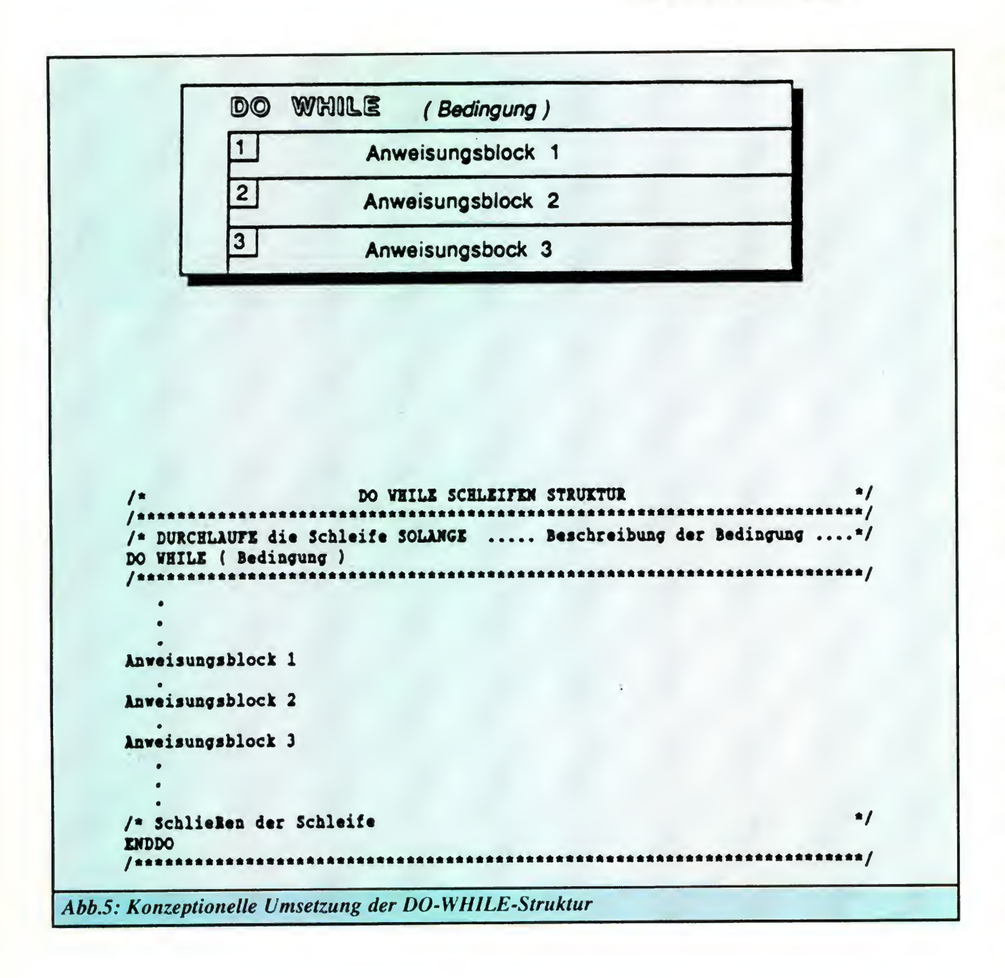


Abb.5: Konzeptionelle Umsetzung der DO-WHILE-Struktur

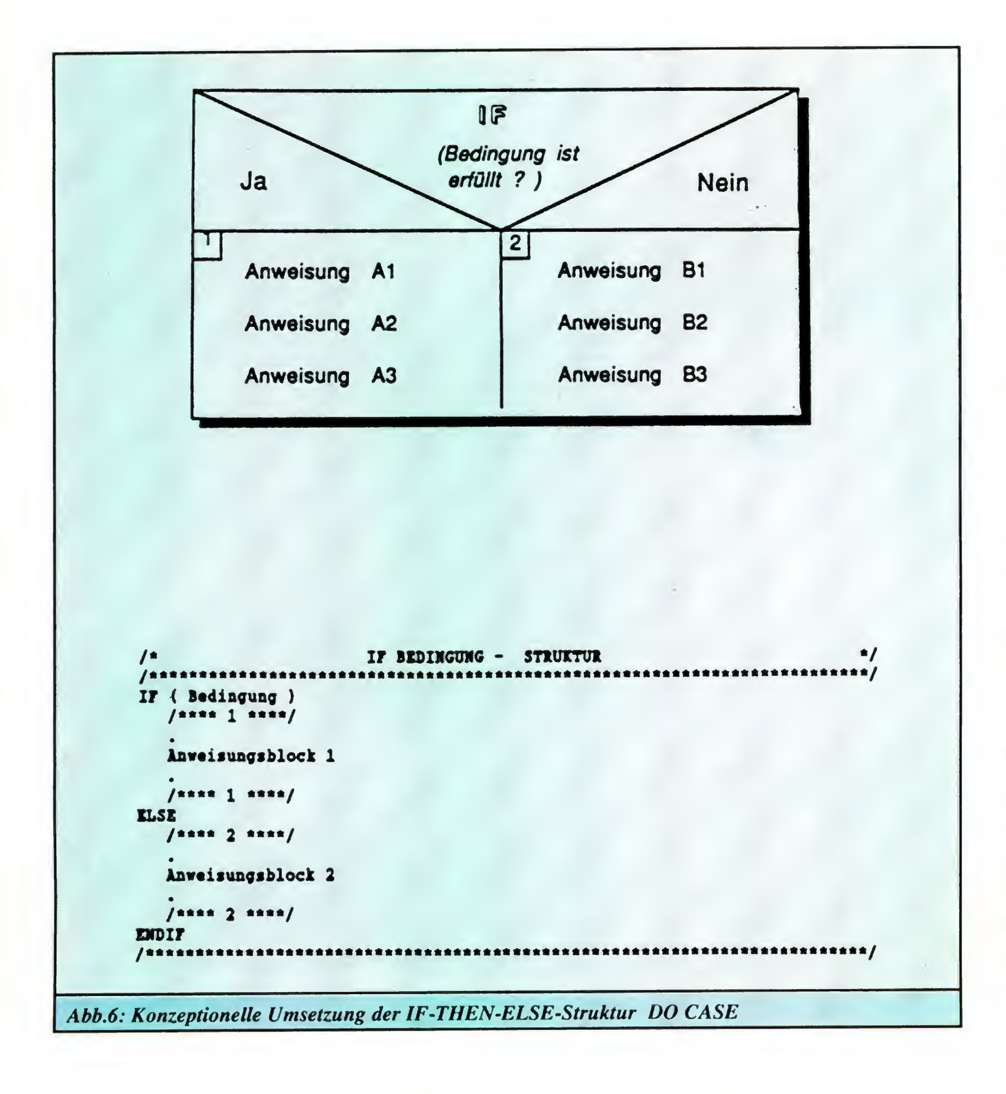


Abb.6: Konzeptionelle Umsetzung der IF-THEN-ELSE-Struktur DO CASE

dessen Bedingung zuerst erfüllt ist. Danach wird die Programmabarbeitung nach dem ENDCASE fortgesetzt.

Trifft keine der Bedingungen zu, kann mit OTHERWISE ein Anweisungsblock angefügt werden, der dann abgearbeitet wird. Ansonsten wird das Programm direkt hinter ENDCASE fortgesetzt (siehe *Abb. 7)*.

Zusammenfassend kann man sagen, daß man mit diesen 4 Programmstrukturen - dem Anweisungsblock, - der DO WHILE-Schleife,- der IF THEN ELSE- und der DO CASE-Bedingung ein Problem vollständig und eindeutig lösen kann.

Ein- und Ausgabe-Befehle

ADIMENS bietet eine umfassende Unterstützung bei der Datenein- und ausgabe. Der einfachste Befehl zur Erzeugung einer Ausgabe wird durch ein Fragezeichen (?) eingeleitet. Das Fragezeichen, gefolgt von einem beliebigen Ausdruck oder einer Folge von Ausdrücken, bewirkt, daß der Wert des Ausdrucks auf dem Bildschirm oder dem Drucker ausgegeben wird.

? < *Ausdruck 1* >

Die Ausgabe erfolgt dabei in der nächsten Zeile. Sollen mehrere Ausdrücke ausgegeben werden, müssen sie durch Kommata voneinander getrennt werden. Soll die Ausgabe direkt in der aktuellen Zeile erfolgen, müssen zwei Fragezeichen als Ausgabebefehl eingegeben werden :

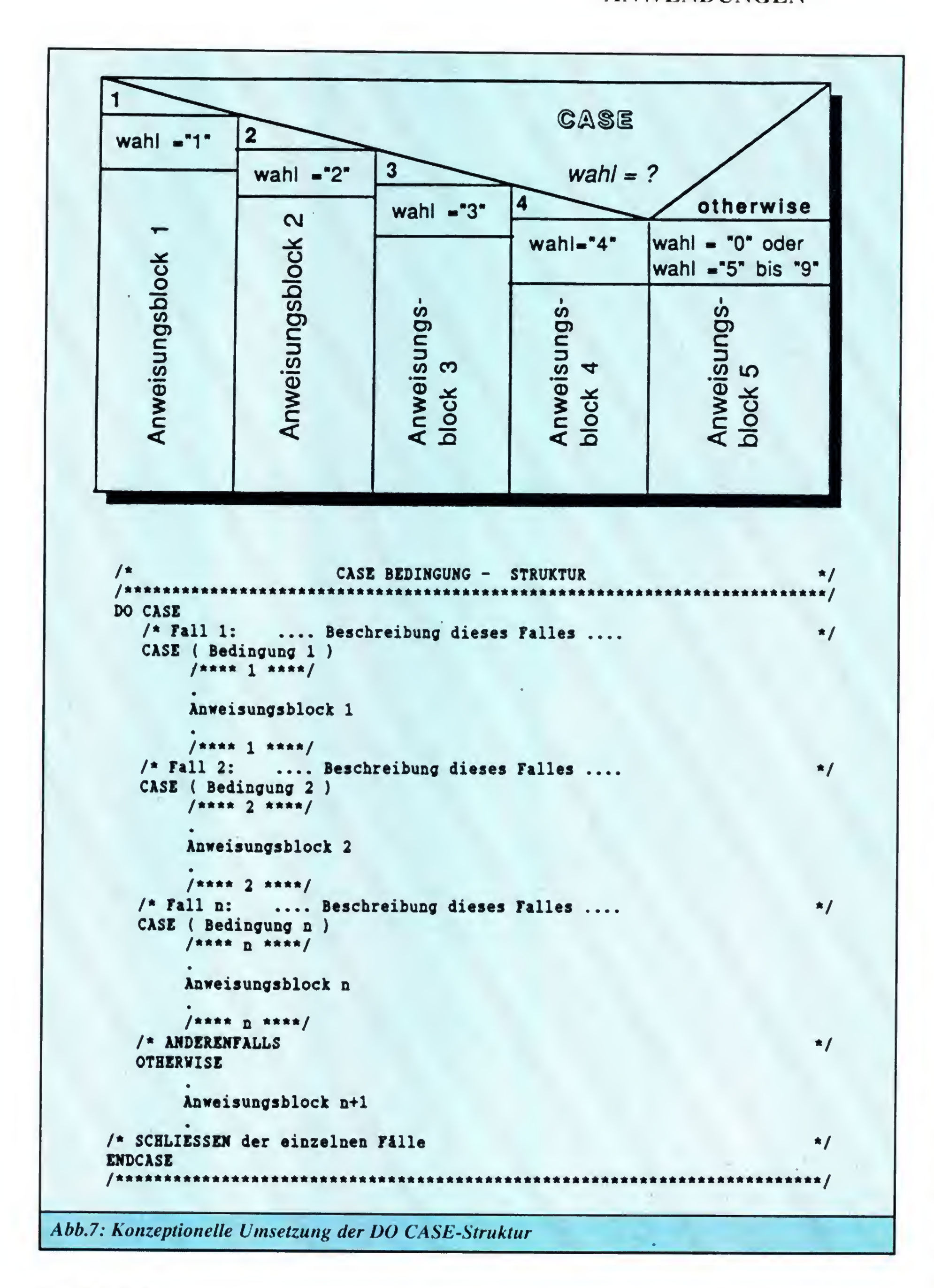
?? < *Ausdruck 2* >

Für die Eingabe eines einzelnen Zeichens dient der Befehl:

WAIT *Information auf dem Bildschirm* **TO** *Variablen Namen*

WAIT unterbricht einen Programmablauf. Das Programm wird erst dann weiter abgearbeitet, wenn entweder ein beliebiges Zeichen eingegeben oder eine beliebige Taste gedrückt wird. Durch die Erweiterung des WAIT-Befehls mit TO Variablen Name wird die Eingabe als Zeichenkette der angegebenen Variablen zugewiesen. Dabei werden die Returntaste und alle weiteren Sondertasten, als leere Sondertasten als leere Zeichenketten betrachtet. Eine zweite Möglichkeit, den WAIT-Befehl zu erweitern, besteht darin, ihm eine Information anzufügen.

geht weiter....




```
/*                      CASE BEDINGUNG -  STRUKTUR                      */
/***********************************************************************/
DO CASE
   /* Fall 1:    .... Beschreibung dieses Falles ....                   */
   CASE ( Bedingung 1 )
        /**** 1 ****/
        .
        Anweisungsblock 1
        .
        /**** 1 ****/
   /* Fall 2:    .... Beschreibung dieses Falles ....                   */
   CASE ( Bedingung 2 )
        /**** 2 ****/
        .
        Anweisungsblock 2
        .
        /**** 2 ****/
   /* Fall n:    .... Beschreibung dieses Falles ....                   */
   CASE ( Bedingung n )
        /**** n ****/
        .
        Anweisungsblock n
        .
        /**** n ****/
   /* ANDERENFALLS                                                      */
   OTHERWISE
        .
        Anweisungsblock n+1
        .
/* SCHLIESSEN der einzelnen Fälle                                      */
ENDCASE
/***********************************************************************/
```

Abb.7: Konzeptionelle Umsetzung der DO CASE-Struktur

Befehle zum Maskenaufbau

Mit TALK ist es möglich, mit den in INIT definierten Masken zu arbeiten, oder sich neue Masken zu erstellen, um sich Datensätze ausgeben zu lassen. Die neuen Masken können entweder Merkmale aus der Datenbank oder Variablen enthalten. Es ist außerdem möglich, eine Maske mit Kommentaren zu versehen. Vor dem Aufbau einer **selbstdefinierten Maske** sollte der Bildschirm gelöscht werden. Das Löschen erfolgt mit Hilfe des CLEAR- Befehls.

An welcher Stelle ein Kommentar oder eine Merkmalsausprägung in der Maske stehen soll, wird über den @ <*Zeile*>,<*Spalte*>-Befehl festgelegt. Die Position in der Maske wird durch zwei Zahlen festgelegt. Die erste Zahl gibt die Zeilennummer, die zweite die Spaltennummer an.

Kommentare werden durch den **SAY** <*Ausdruck*> in eine Maske eingefügt. Neben Kommentaren können auch ganze Ausdrücke am Bildschirm ausgegeben werden. Die Ausgabe ist jedoch auf einen Ausdruck beschränkt. Der SAY-Befehl arbeitet also ähnlich wie das doppelte Fragezeichen und kann auch dementsprechend eingesetzt werden. Während der SAY-Befehl dazu dient, eine Maske optisch zu gestalten, lassen sich mit dem **GET**-Befehl **Merkmalsausprägungen** oder **Variableinhalte einer Maske** bearbeiten.

GET <*Variablenname*>

PICTURE <*Kontroll-Anweisung*>

Mit PICTURE kann der Inhalt eines Merkmals durch Kommentare oder Ergänzungen erweitert oder nur eine teilweise Bearbeitung zugelassen werden.

Alle GET-Anweisungen innerhalb eines Programmes werden zunächst intern zwischengespeichert. Das Abarbeiten der GET-Anweisungen erfolgt erst über einen gesonderten Befehl **READ.**

Diskussion des Menüaufbaues

Ein einfacher Menüaufbau könnte folgendermaßen aussehen :

```
CLEAR
? " .............. "
? " .............. "
? " .............. "
usw.
```

Ein Nachteil ist dabei aber, daß bei später gewünschter Verschiebung der Menüdarstellung sehr viele Leerschritte von Hand eingefügt oder gelöscht werden müssen. Abhilfe ergibt sich bei **direkter Adressierung.**

```
CLEAR
@ 10,20 SAY " 0 ... Programmende"

@ 11,20 SAY " 1 ... Stammdatenver-
                    waltung "
@ 12,20 SAY " 2 .... Organisation "
```

Hier müßten zwar evtl. nur noch Zahlen geändert werden, aber immer noch mehrere. So müßten z.B. bei einer gewünschten Verschiebung um 10 Spalten nach rechts alle "20" durch "30" ersetzt werden. Man kann nun die Tabulatoren durch eine Variable ersetzen. Zum Beispiel durch TAB = 20.

In **Abb. 8 oben** wurde ein allgemeines Struktogramm für eine Menüsteuerung entworfen. Die Auswahl der Untermenüpunkte erfolgt durch die Zuweisung der Variablen wahl. Sie kann die Werte von 1 bis n annehmen. Dabei ist n die Zahl der möglichen aufzurufenden Module. Es wird dann vereinbart, daß bei der Eingabe der Zahl 0 das Programm beendet werden soll.

Zum Standardumfang eines Menüprogrammes gehört außerdem eine sogenannte **Plausibilitätsprüfung** (bzw. -kontrolle), d.h. eine Abfrage, die nicht erlaubte Eingaben erkennen und zurückweisen soll. Da dazu eventuell eine mehrfache Wiederholung der Eingabe möglich sein muß, ist im Programm eine zweite ("innere") Schleife erforderlich. Bei nä-

wahl = " "
WHILE wahl <> "0"
Menü Ausgabe
WHILE wahl <"0" oder wahl >"n"
Eingabe wahl
CASE wahl =?
otherw. | 1 | 2 | 3 | n
Fehlermeldung | DO Programm Untermodul 1 | DO Programm Untermodul 2 | DO Programm Untermodul 3 | DO Programm Untermodul n

wahl = " "
WHILE wahl <> "0"
Menü Ausgabe Speichern wahl = " "
WHILE wahl <"0" oder wahl >"n"
Eingabe wahl
CASE wahl =?
otherw. | 0 | 1 | 2 | n
Fehlermeldung | | DO Programm Untermodul 1 | DO Programm Untermodul 2 | DO Programm Untermodul n

Abb.8: Entwurfsalternativen beim Menüaufbau

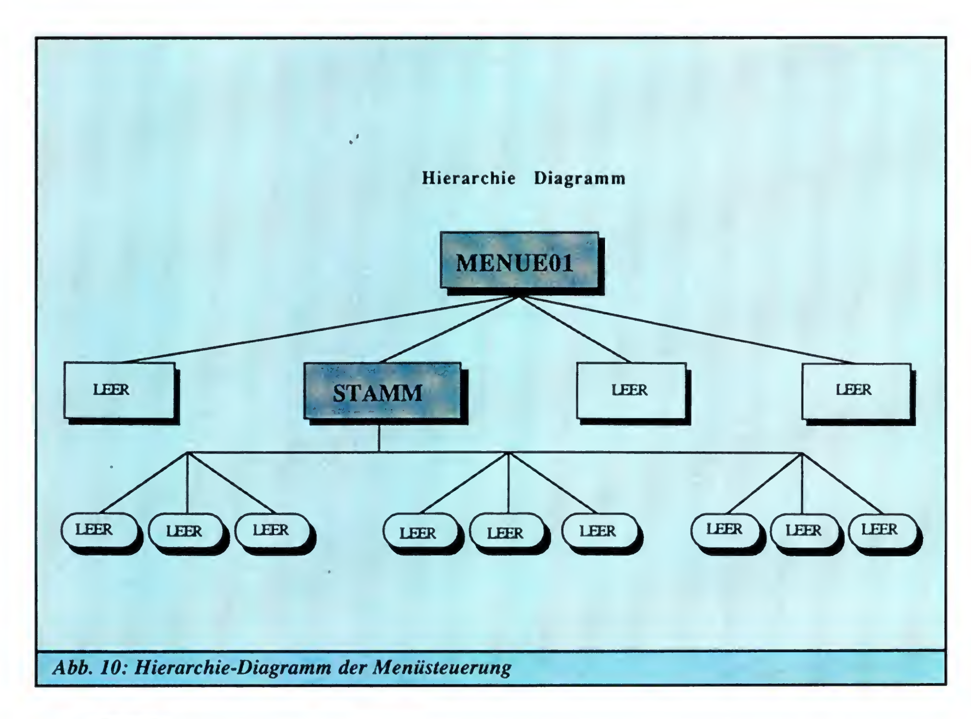


Abb. 10: Hierarchie-Diagramm der Menüsteuerung

herer Betrachtung dieses Algorithmus' zeigen sich jedoch zwei Fehler:

- Bei Eingabe "0" erfolgt eine Fehlermeldung durch die Plausibilitätsprüfung (weil "0" durch "otherwise") erfaßt wurde).
- Nach erlaubter Eingabe im 1. Durchgang (z.B. "1") erfolgt ein unendlicher Menüdurchlauf, da mit wahl = "1" ab dem 2. Durchgang kein Eintritt in die innere Schleife mehr möglich ist.

Deshalb wurde **folgende Abhilfe** geschaffen.

- Der Fall CASE wahl ="0" wird in die CASE-Anweisung mit aufgenommen, jedoch ohne Befehlsinhalt. Dadurch wird der Fall wahl ="0" nicht mehr von "otherwise" erfaßt.
- Die Anweisung wahl =" " (Leerzeichen) wird (wie bei der äußeren Schleife) vor die WHILE-Anweisung gesetzt.

Daraus ergibt sich das Struktogramm in **Abb. 8 unten.** Dieses Schema wird bei allen weiteren Menüauswahlen weiterverwendet.

Die **Feinstruktur** des Hauptmenüs eines möglichen Anwendungsprogrammes wird in **Abb. 9** gezeigt.

Im **Anweisungsblock 1** werden die Schalterstellungen festgelegt. Dabei wird immer am Anfang der Bildschirm gelöscht und die automatische Maskenanzeige unterdrückt. Dies hat den Vorteil, daß, wenn man eine Datenbank mit einer zugehörigen Datei öffnet, die Maske mit der ersten Datei nicht angezeigt wird. Das automatische Löschen des Bildschirmes wird unterdrückt, und die Sonderregelungen für die Funktion FIND werden abgeschaltet.

Im **Anweisungsblock 2** werden die *globalen Variablen* festgelegt. Globale Variablen sind im ganzen Programm gültig. Bei ADIMENS Talk wird dies dadurch geregelt, daß die Variablen in einem Unterprogramm nicht deklariert werden müssen. Beim Übersetzen des Unterprogrammes wird dann diese Variable importiert und als global betrachtet.

Nach diesen Systemeinstellungen wird die DO WHILE-Schleife so lange ausgeführt, bis der Benutzer mit der Eingabe "0" das Programmende selbst definiert. Die Eingabe wird der Variablen "wahl" zugeordnet, welche auf den Wert " " (1

Leerzeichen) oder äquivalent dazu SPACE(1) gesetzt wurde. Innerhalb dieser Schleife wird der Anweisungsblock 3 abgearbeitet, der die Menüausgabe auf dem Bildschirm anzeigt. Er wurde in die DO-WHILE-Schleife gesetzt, damit bei jeder Änderung auch die Uhrzeit und das Datum korrekt angezeigt werden. Aus dem Struktogramm in **Abb. 9.** wird ersichtlich, daß nun die Variable "wahl" erneut auf den Wert SPACE (1) gesetzt wird, um nicht in eine Endlos-Schleife zu laufen. Die Schleife für die Plausibilitätskontrolle erlaubt nur die Eingaben für die Variable "wahl" von 0 bis 4. Die Masken werden mit den SAY-Kommandos aufgebaut. Mit dem Kommando WAIT wird eine **Eingabe von der Tastatur** erwartet, und ein **Tastendruck** wird der Variablen "wahl" zugewiesen.

Im **Anweisungsblock 4** findet eine DO CASE-Abfrage statt, die bewirkt, daß die Variable "wahl" auf die Werte 0 bis 4 abgefragt wird. Je nachdem, welche Ziffer eingegeben wurde, wird mit DO das entsprechendUntermodul ausgeführt. Die DO CASE-Schleife wird mit ENDCASE abgeschlossen. Die DO WHILE-Schleife wird mit ENDDO abgeschlossen.Listing 1 und Listing 2 zeigen das durchgängige Konzept einer Menüsteuerung. Im Listing 2 findet eine differenzierte Menüauswahl statt.Zum Schreiben, Ändern und Speichern des Quelltextes wird unabhängig von der Programmiersprache ADIMENS Talk ein sogenannter Editor benötigt. Dies kann im Prinzip ein beliebiges Textverarbeitungsprogramm sein, jedoch mit folgenden Einschränkungen:

- Das Programm muß über einen " non-document-mode" verfügen, d.h., es darf keine Steuerzeichen in den Programmtext einfügen.
- Es sollte die deutschen Umlaute auf gleiche Weise darstellen, wie die eigentliche Programmiersprache.

Für die glücklichen Atari-Besitzer mit 1 Megabyte RAM eignet sich eine RAM-DISK. Für die Programmierung der Menüsteuerung müssen daher nur die Programme TALK.PRG, TALK.RSC und ihr "Liebingseditor" in die Ram Disk geladen werden.Die erstellten Quelltexte haben die Kennzeichnung *. TLK und die dann mit ADI Talk übersetzten Programme *.TLP. Wollen Sie nun selbständig eine Menüsteuerung generieren, so empfiehlt es sich, sich ein einheitliches Modulkonzept zurechtzulegen. Das einzige, was sie bei der vorgestellten Menüsteuerung noch ändern müßten, ist der Aufbau der einzelnen Auswahlpunkte.

Dieses Modul dient zur Menüsteuerung. Dabei hat man 4 Möglichkeiten zur Auswahl in verschiedene Untermoduln zu springen.

1 Festlegen der Schalterstellungen / Löschen des Bildschirmes
2 globale Variablendeklaration :
Speichere den Wert : "F" in CARRY ,SPACE(1) in Wahl1
SPACE(2) in Wahl 2 und in Wahl 3
DO WHILE (der Benutzer das Programm verlassen will)
3 Menü-Ausgabe auf dem Bildschirm mit SAY - Befehl
4 lokale Variablendekl.: SPACE(1) in Wahl, ROW()+4 in Zeile
DO WHILE nicht einen Wert zwischen 0 und 4 eingeben
5 EINGABE wahl WAIT-Befehl
CASE
wahl = 0 bis 4 ?
otherwise
0
1
2
3
4
Erzeugen einer Falschmeldung mit Warnton
Untermenü Stammdaten
Untermenü Einkauf
Untermenü Organisation
Untermenü Statistik

Abb. 9: Feinstruktur einer Menüauswahl

Schluß, Zusammenfassung und Ausblick

Im 1. Teil der flexiblen Modulprogrammierung haben Sie die wichtigsten Programmstrukturen kennengelernt, die eine Anwendungsprogrammierung mit ADIMENS Talk erlauben. Danach wurde im Quelltext näher auf die Möglichkeiten einer strukturierten Programmierung, den TOP-DOWN-Entwurf und die Bedeutung der Struktogramme und deren konzeptionelle Umsetzung eingegangen. Nach einer Stoffsammlung an Teilaufgaben konnte die hierarchische Gliederung eines beispielhaften Menüaufbaues vorgestellt werden.Mit dieser flexiblen und universell einsetzbaren Menüsteuerung können Sie ihr Anwendungprogramm gestalten.Im Teil2 wird gezeigt werden, wie man mit einem flexiblen Modul Merkmale Ihrer Datenbank, die in verschiedenen Dateien vorkommen, benutzerfreundlich erfassen, löschen und abspeichern kann.

Hans-Ulrich Mayer

/1/ *Günter Spur, Frank-Lothar Krause, CAD-Technik, Lehr- und Arbeitsbuch für die Rechnerunterstützung in Konstruktion und Arbeitsplanung, Hanser Verlag.*

```
  1:  /*                    PROGRAMM MENUE01.TLK                   */
  2: /*********************************************************/
  3:  /*    Dies ist ein Beispiel für eine Menüsteuerung         */
  4:  /*       (C) copyright by Hans-Ulrich Mayer  1988          */
  5: /*********************************************************/
  6:
  7:  /**** 1 ****/
  8:  /* FESTLEGUNG DER SCHALTERSTELLUNGEN                        *
  9:  /* Löschen des Bildschirmes                                 *
 10:  CLEAR
 11:  /* Unterdrücken der automatischen Maskenanzeige             *
 12:  SET AUTOSC OFF
 13:  /* Die Sonderregelungen für die Funktion FIND sind          *
 14:  /* abgeschaltet                                             *
 15:  SET DBASE OFF
 16:  /**** 1 ****/
 17:
 18: /**********************************************************/
 19:  /* DURCHLAUFE die Schleife SOLANGE der Benutzer das       */
 20:  /* Programm durch die Eingabe "0" verlassen will.         */
 21:  DO WHILE ( WAHL <> "0" )
 22: /**********************************************************/
 23:     /**** 3 ****/
 24:     /* MENÜAUSGABE - Aufbau der Kopfzeile                  */
 25:     CLEAR
 26:     /* aktuelle Datumsanzeige und der Uhrzeit              */
 27:     @ 02     ,00  SAY " Datum "
 28:     @ 03     ,00  SAY DATE()
 29:     @ 02     ,72  SAY "Uhrzeit"
 30:     @ 03     ,72  SAY TIME()
 31:     @ 01     ,00  SAY BILD1 + BILD1
 32:     @ ROW()  ,TAB SAY "          MENÜ STEUERUNG
 33:                                          *** V 1.01 *** "
 34:   @ ROW()+1,TAB SAY "flexible Anwendungsprogrammierung
 35:                                    mit ADIMENS Talk "
 36:    @ ROW()+1,00  SAY BILD2 + BILD2
 37:
 38:    /* MENÜAUSGABE - Aufbau d.einzelnen Auswahlpunkte */
 39:
 40:     @ ROW()+1,TAB SAY "  **** H A U P T M E N U E **** "
 41:      @ ROW()+1,TAB SAY "                                 "
 42:     @ ROW()+1, 00 SAY BILD1 + BILD1
 43:     @ ROW()+1,TAB SAY " 0 ..... PROGRAMM ENDE           "
 44:      @ ROW()+1,TAB SAY "                                "
 45:      @ ROW()+1,TAB SAY "                                "
 46:     @ ROW()+1,TAB SAY " 1 ..... S T A M M D A T E N     "
 47:      @ ROW()+1,TAB SAY "                                "
 48:     @ ROW()+1,TAB SAY " 2 ..... E I N K A U F           "
 49:      @ ROW()+1,TAB SAY "                                "
 50:     @ ROW()+1,TAB SAY " 3 ..... O R G A N I S A T I O N"
 51:      @ ROW()+1,TAB SAY "                                "
 52:     @ ROW()+1,TAB SAY " 4 ..... S T A T I S T I K       "
 53:      @ ROW()+1,TAB SAY "                                "
 54:     @ ROW()+1, 00 SAY BILD1 + BILD1
 55:     /**** 3 ****/
 56:
 57:   /************************************************/
 58:
 59:     /**** 4 ****/
 60:     /* LOKALE VARIABLENDEKLARATION - die Variablen    */
 61:     /* gelten für dieses Programm                     */
 62:     STORE SPACE(1) TO WAHL
 63:     STORE ROW()+4  TO ZEILE
 64:     /**** 4 ****/
 65:
 66:  /*************************************************/
 67:    /* DURCHLAUFE d.Schleife SOLANGE ein falscher Wert */
 68:     /* eingegeben wird                                */
 69:     DO WHILE ( (WAHL < "0") OR (WAHL > "4") )
 70:  /*************************************************/
 71:
 72:        /**** 5 ****/
 73:        /* EINGABE WAHL                                */
 74:       @ ZEILE, 00 WAIT " Bitte wählen Sie .... " TO WAHL
 75:        /**** 5 ****/
 76:
 77:      /**********************************************/
 78:       /* FALLUNTERSCHEIDUNG  - Verzw.in Untermoduln   */
 79:        DO CASE
 80:
 81:           /* FALL 0 : Programm beenden und Zurück auf  */
 82:            /* Betriebssystemebene                       */
 83:            CASE WAHL = "0"
 84:             /* nichts tun                               */
 85:           /* FALL 1 : Sprung ins Untermenü STAMMDATEN  */
 86:            CASE WAHL = "1"
 87:                  DO "STAMM"
 88:           /* FALL 2 : Sprung ins Untermenü EINKAUF     */
 89:            CASE WAHL = "2"
 90:                  DO "LEER"
 91:           /* FALL 3 : Sprung ins Untermenü ORGANISATION*/
 92:            CASE WAHL = "3"
 93:                  DO "LEER"
 94:           /* FALL 4 : Sprung ins Untermenü STATISTIK   */
 95:            CASE WAHL = "4"
 96:                  DO "LEER"
 97:            /* ANDERENFALLS                              */
 98:            OTHERWISE
 99:              /* Erzeugen einer Falschmeldung m.Warnton */
100:               BELL
101:               @ 22,19 SAY "Falsche Eingabe !!!  Bitte
102:                                        wiederholen ..."
103:
104:        ENDCASE
105:      /**********************************************/
106:
107:    /* Beenden der zweiten Schleife, da ein richtiger */
108:     /* Wert eingegeben wurde                         */
109:    ENDDO
110:  /**************************************************/
111:
112:  /* Beenden der ersten Schleife da der Benutzer das   */
113:  /* Programm verlassen will                           */
114:  ENDDO
115: /****************************************************/
```

Listing 1: Hauptmenü

```
  1:  /*                 PROGRAMM STAMM.TLK                     */
  2: /*********************************************************/
  3:  /*    Dies ist ein Beispiel für eine Menüsteuerung        */
  4:  /*      (C) copyright by Hans-Ulrich Mayer 1988           */
  5: /*********************************************************/
  6:
  7:  /**** 1 ****/
  8:  /* FESTLEGUNG DER SCHALTERSTELLUNGEN                      */
  9:  /* Löschen des Bildschirmes                               */
 10:  CLEAR
 11:  /* Unterdrücken der automatischen Maskenanzeige           */
 12:  SET AUTOSC OFF
 13:  /* Die Sonderregelungen für die Funktion FIND sind        */
 14:  /* abgeschaltet                                           */
 15:  SET DBASE OFF
 16:  /**** 1 ****/
 17:
 18: /**********************************************************/
 19:  /* DURCHLAUFE die Schleife SOLANGE der Benutzer das       */
 20:  /* Programm durch die Eingabe "0" verlassen will.         */
 21:  DO WHILE ( WAHL <> "0" )
 22: /**********************************************************/
 23:     /**** 2 ****/
 24:     /* MENÜAUSGABE - Aufbau der Kopfzeile                  */
 25:     CLEAR
 26:     /* aktuelle Datumsanzeige und der Uhrzeit              */
 27:     @ 02     ,00  SAY " Datum "
 28:     @ 03     ,00  SAY DATE()
 29:     @ 02     ,72  SAY "Uhrzeit"
 30:     @ 03     ,72  SAY TIME()
 31:     @ 01     ,00  SAY BILD1 + BILD1
 32:     @ ROW()  ,TAB SAY "          MENÜ STEUERUNG
 33:                                         *** V 1.01 *** "
 34:   @ ROW()+1,TAB SAY "flexible Anwendungsprogrammierung
 35:                                     mit ADIMENS Talk"
 36:     @ ROW()+1,00  SAY BILD2 + BILD2
 37:
 38:    /* MENÜAUSGABE - Aufbau d.einzelnen Auswahlpunkte */
 39:
 40:     @ ROW()+1,TAB SAY " **** S T A M M D A T E N **** "
 41:     @ ROW()+1, 00 SAY BILD1 + BILD1
 42:     @ ROW()+1,TAB SAY " 0 ...... ZURÜCK  zum Hauptmenü "
 43:      @ ROW()+1,TAB SAY "                                "
 44:      @ ROW()+1,TAB SAY "                                "
 45:     @ ROW()+1,TAB SAY "      L I E F E R A N T E N     "
 46:     @ ROW()+1,TAB SAY " 1... Erfassen    2... Ändern
```

Listing geht weiter...

```
 47:                                         3... Suchen   "
 48:      @ ROW()+1,TAB SAY "                              "
 49:      @ ROW()+1,TAB SAY "              K U N D E N     "
 50:     @ ROW()+1,TAB SAY " 4... Erfassen     5... Ändern
 51:                                         6... Suchen   "
 52:      @ ROW()+1,TAB SAY "                              "
 53:      @ ROW()+1,TAB SAY "     M I T A R B E I T E R    "
 54:     @ ROW()+1,TAB SAY " 7... Erfassen     8...Ändern
 55:                                          9...Suchen   "
 56:     @ ROW()+1,TAB SAY "
 57:     @ ROW()+1, 00 SAY BILD1 + BILD1
 58:     /**** 2 ****/
 59:
 60:  /**************************************************/
 61:
 62:     /**** 3 ****/
 63:     /* LOKALE VARIABLENDEKLARATION - die Variablen     */
 64:     /* gelten für dieses Programm                      */
 65:     STORE SPACE(1) TO WAHL
 66:     STORE ROW()+4  TO ZEILE
 67:     /**** 3 ****/
 68:
 69:  /**************************************************/
 70:     /* DURCHLAUFE die Schleife SOLANGE ein falscher    */
 71:     /* Wert eingegeben wird                            */
 72:     DO WHILE ( (WAHL < "0") OR (WAHL > "9") )
 73:  /**************************************************/
 74:
 75:        /**** 4 ****/
 76:        /* EINGABE WAHL                                 */
 77:       @ ZEILE, 00 WAIT " Bitte wählen Sie .... " TO WAHL
 78:        /**** 4 ****/
 79:
 80:      /**************************************************/
 81:       /* FALLUNTERSCHEIDUNG - Verzw.in Untermoduln     */
 82:       DO CASE
 83:
 84:          /* FALL 0 : Programm beenden und zurück zum   */
 85:           /*Hauptmenü                                 */
 86:          CASE WAHL = "0"
 87:          /* nichts tun */
 88:         /* FALL 1 : Sprung ins Untermenü LIEFERANTEN */
 89:          /* ERFASSEN                                  */
 90:          CASE WAHL = "1"
 91:               DO "LERFASSE"
 92:         /* FALL 1 : Sprung ins Untermenü LIEFERANTEN */
 93:          /* ÄNDERN                                    */
 94:          CASE WAHL = "2"
 95:               DO "LEER"
 96:         /* FALL 1 : Sprung ins Untermenü LIEFERANTEN */
 97:          /* SUCHEN                                    */
 98:          CASE WAHL = "3"
 99:               DO "LEER"
100:         /* FALL 1 : Sprung ins Untermenü KUNDEN      */
101:          /* ERFASSEN                                  */
102:          CASE WAHL = "4"
103:               DO "LEER"
104:         /* FALL 1 : Sprung ins Untermenü KUNDEN      */
105:          /* ÄNDERN                                    */
106:          CASE WAHL = "5"
107:               DO "LEER"
108:         /* FALL 1 : Sprung ins Untermenü KUNDEN      */
109:          /* SUCHEN                                    */
110:          CASE WAHL = "6"
111:               DO "LEER"
112:         /* FALL 1 : Sprung ins Untermenü MITARBEITER */
113:          /* ERFASSEN                                  */
114:          CASE WAHL = "7"
115:               DO "LEER"
116:         /* FALL 1 : Sprung ins Untermenü MITARBEITER */
117:          /* ÄNDERN                                    */
118:          CASE WAHL = "8"
119:               DO "LEER02"
120:         /* FALL 1 : Sprung ins Untermenü MITARBEITER */
121:          /* SUCHEN                                    */
122:          CASE WAHL = "9"
123:               DO "LEER"
124:
125:          /* ANDERENFALLS                              */
126:          OTHERWISE
127:         /* Erzeugen einer Falschmeldung mit Warnton  */
128:             BELL
129:             @ 22,19 SAY "Falsche Eingabe !!!  Bitte
130:                                         wiederholen ... "
131:
132:       ENDCASE
133:     /**************************************************/
134:
135:    /* Beenden der zweiten Schleife, da ein richtiger */
136:     /* Wert eingegeben wurde                          */
137:    ENDDO
138:  /**************************************************/
139:
140:  /* Beenden der ersten Schleife, da der Benutzer zum  */
141:  /* Hauptmenü will.                                   */
142:  ENDDO
143:  /**************************************************/
144:
145:  /* Rücksprung ins Hauptmenü                          */
146:  STORE "1" TO WAHL
147:  DO "MENUE01"
148:  /**************************************************/
```

Listing 2: Untermenü

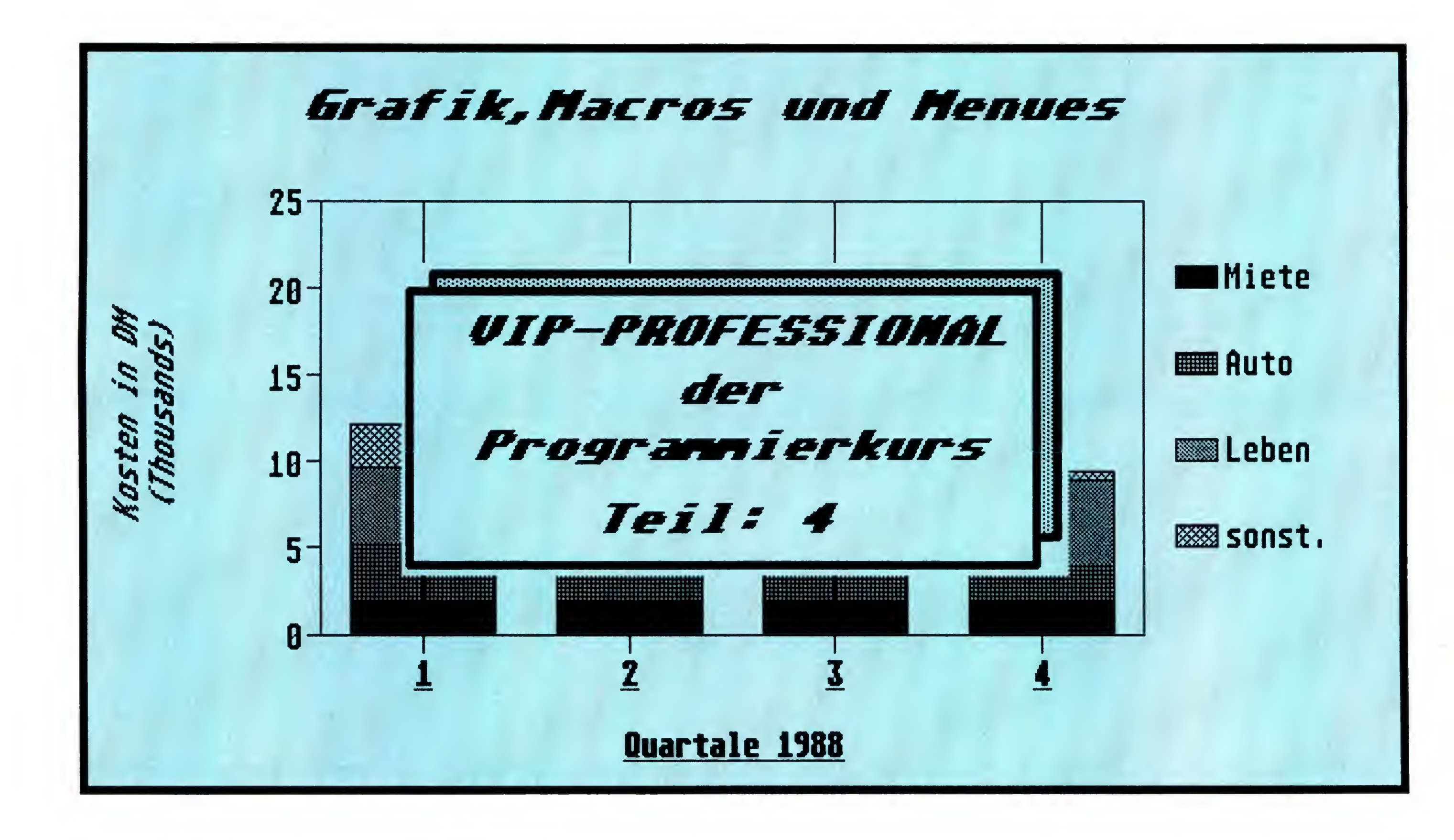
```
 1:  /*                  PROGRAMM LEER.TLK                  */
 2: /*****************************************************/
 3:  /*      Dies ist ein Beispiel eines leeren Moduls     */
 4:  /*      (C) copyright by Hans-Ulrich Mayer   1988     */
 5: /*****************************************************/
 6:
 7:  /**** 1 ****/
 8:  /* FESTLEGUNG DER SCHALTERSTELLUNGEN                  */
 9:  /* Löschen des Bildschirmes                           */
10:  CLEAR
11:  /* Unterdrücken der automatischen Maskenanzeige       */
12:  SET AUTOSC OFF
13:  /* Die Sonderregelungen für die Funktion FIND sind    */
14:  /* abgeschaltet                                       */
15:  SET DBASE OFF
16:  /**** 1 ****/
17:
18: /*****************************************************/
19:
20:  /**** 2 ****/
21:  /* MENÜAUSGABE - Aufbau der Kopfzeile                 */
22:  CLEAR
23:  /* aktuelle Datumsanzeige und Uhrzeit                 */
24:  @ 02,00 SAY " DATUM "
25:  @ 03,00 SAY DATE()
26:  @ 02,72 SAY "Uhrzeit"
27:  @ 03,72 SAY TIME()
28:  @ 01,00 SAY BILD1 + BILD1
29:  @ ROW(),TAB SAY "     MENÜ STEUERUNG  *** V 1.01 ***   "
30: @ ROW()+1,TAB SAY "flexible Anwendungsprogrammierung
31:                                      mit ADIMENS Talk"
32:  @ ROW()+1,00 SAY BILD2 + BILD2
33:
34:  /* BILDSCHIRM ANZEIGE - Meldung auf dem Bildschirm    */
35:
36:   @ ROW()+1,TAB SAY "      **** Leeres Modul ****      "
37:   @ ROW()+1,TAB SAY "                                  "
38:  @ ROW()+1,00 SAY BILD1 + BILD1
39:   @ ROW()+1,TAB SAY "                                  "
40:   @ ROW()+1,TAB SAY "                                  "
```

Listing geht weiter...

```
41:  @ ROW()+1,TAB SAY "                                    "
42:  @ ROW()+1,TAB SAY "  Leider ist dieses Programm noch "
43:  @ ROW()+1,TAB SAY "                                    "
44:  @ ROW()+1,TAB SAY "  nicht erstellt worden.           "
45:  @ ROW()+1,TAB SAY "                                    "
46:  @ ROW()+1,TAB SAY "                                    "
47:  @ ROW()+1,TAB SAY " Bitte drücken Sie eine beliebige "
48:  @ ROW()+1,TAB SAY " Taste, um wieder ins vorherige   "
49:  @ ROW()+1,TAB SAY " Menü zurückzukehren              "
50:  @ ROW()+1,TAB SAY "                                    "
51:  @ ROW()+1,00 SAY BILD1 + BILD1
52:  /**** 2 ****/
53:
54: /*********************************************************/
55:
56:  /**** 3 ****/
57:  /* WARTEN bis eine beliebige Taste gedrückt wurde      */
58:  @ ROW()+4,00 WAIT
59:  /**** 3 ****/
60:
61: /*********************************************************/
62:
63:  /* Rücksprung ins vorherige Menü, von dem aus          */
64:  /* verzweigt wurde.                                    */
```

Listing 3: Leerprogramm

ENDE



Es ist wieder eine schöne Zeit seit unserem letzten Kursteil vergangen. Ich hoffe, Sie haben sich in der Macrowelt ein wenig ausgetobt und konnten mit meinen Literaturhinweisen die eine oder andere Problemstellung nachschlagen und lösen. Wie bereits angekündigt, wollen wir uns heute der grafischen Darstellung von Zahlenwerten nähern. Oft wundert sich der Laie, wenn alle Jahre wieder bei den Hochrechnungen zu Regierungswahlen die tollen Charts auf dem Fernsehschirm erscheinen. Die Säulen und Kreisdiagramme sind ja nunmalsehr leicht zu verstehen, doch wer malt die denn so schön und so schnell? Natürlich ein Kalkulationsprogramm, das in der Lage ist, aus Werten einer Tabelle Zeichnungen zu generieren.

Unser VIP-Professional besitzt auch einen kleinen Grafikteil, der es ermöglicht, numerische Daten, die in ein Arbeitsblatt eingegeben werden, grafisch darzustellen. VIP kennt fünf verschiedene Arten von Diagrammen: Liniendiagramme, Balkendiagramme, XY-Diagramme, gestaffelt Balkendiagramme und Kreisdiagramme.

Mit Liniendiagrammen werden Änderungen der Daten im Laufe der Zeit am besten dargestellt. In dem Beispiel (Abbildung 1) könnte man den Kostenverlauf der Quartale 1 bis 4 wählen.

Mit Balkendiagrammen können Unterschiede zwischen Datenelementen besonders gut hervorgehoben werden, während mit xy-Diagrammen die Beziehungen zwischen zwei Datengruppen dargestellt werden können. Mit Kreisdiagrammen lassen sich Teile eines Ganzen vergleichen.

Bevor wir mit unserer Grafik-Arbeit beginnen, bauen Sie sich bitte Ihr Arbeitsblatt gemäß Abbildung 1 auf.

Es gibt immer ein erstes Mal

Es ist meiner Ansicht nach die Zeit gekommen, daß wir uns von meinem, für Sie recht bequemen, Vormachen trennen müssen. Ab sofort sprechen wir Fraktur, was heißt, daß unsere Tastatursequenzen gleich in die Macrosprache umgesetzt werden. Meine Abbildung 2 zeigt Ihnen den ersten Grafikmacro mit dem Namen **\G.**

Wenn Sie den Macro sauber abgetippt haben, erinnern Sie sich bitte an unsere Taufsequenz. Also Cursor auf **H3** und **/RNC\G~~.**

(Befehlsleiste, Range, Name, Create,\G, Return,Return)

Unser Macro wurde generiert und kann mit der Tastenkombination **ALT+G** gestartet werden. Doch VIPs kennen bereits meinen Rat: "Zuerst sichern,dann weiterarbeiten !"

Überprüfen Sie bitte, ob Ihre Grafik genauso wie Abbildung 3 aussieht.

Sie sehen, daß es doch recht leicht ist, eine übersichtliche Grafik des Kostenverlaufes aufzubauen. Diese Grafik ließe sich ohne größereSchwierigkeiten in Wordplus einbauen. Damit wir aber ein automatisches Bildabspeichern für die Weiterverarbeitung erreichen,ergänzen wir unseren Macro ab Zeile H19 mit der Grafikgenerierungssequenz: **GNCKURS4~SKURS4~RQ/XQ~.**

(Befehlsleiste, Graph, Name, Create, Kurs 4, Return, Save, Kurs 4, Return, Replace, Quit, Macro-Ende mit /XQ~)

Die Zelle **H19** muß jetzt aber so aussehen: **QVQ** (Quit,View,Quit). Bitte beachten Sie unbedingt, daß Ihr Arbeitsblatt nach dieser Grafik-Generierungssequenz abgespeichert werden muß. VIP benötigt für das spätere Aufrufen einer Grafik ohne Generierungsprozedur unbedingt die Informationen der vorgegebenen Grafikdaten.

VIP und WORDPLUS

Die recht guten Möglichkeiten mit dem

Textverarbeitungsprogramm Wordplus und der Snapshot.ACC, grafische Abbildungen in einen Text zu übernehmen, reizen sehr, die Dokumentationen für Schule und Beruf etwas abwechslungsreicher zu gestalten. Zunächst setze ich voraus, daß Sie ein abgespeichertes File mit dem Namen **KURS4.GPH**, also der Grafik- Extension **GHP** angelegt haben. Dieses Bildformat kann nur mit dem Zusatzprogramm PRINTGRAPH auf der original Vip-Professional-Diskette gelesen werden. In PRINTGRAPH können noch Veränderungen an der Grafikbeschriftung durchgeführt werden. Das Sichern eines geladenen Bildes kann hieraus auch im Degas-Format erfolgen, was für den Mal- und Zeichenkünstler, der dieser Grafik in degasfähigen Zeichenprogrammen noch den letzten Schliff geben möchte, von großer Bedeutung sein kann. Außerdem haben wir in PRINTGRAPH unser GEM-DESKTOP zurückerhalten und damit auch die Möglichkeit, unser Wordplus-ACC-File Snapshot zu starten. Verkleinern und verändern Sie Ihre Grafik nach Belieben, starten Sie Snapshot und definieren Sie den gewünschten Ausschnittbereich. Der Aufforderung nach einer Namensgebung mit der Extension ?????.IMG leisten Sie Folge und speichern das sogenannte GEM-Image-File auf derselben Pfad-Ebene wie später auch Ihr Dokument. Innerhalb von Wordplus stellen Sie den Grafik-Modus an und lesen Ihr Bild ein.

Das Positionieren geht recht einfach, indem Sie mit dem Mauspfeil das Bild berühren und durch Gedrückthalten der linken Maustaste verschieben. Ihr VIP-WORDPLUS Dokument können Sie sichern und später mit Text und Grafik ausdrucken. Je nach Qualität des Druckertreibers und Druckers können sich die meisten Dokumentationen recht gut sehen lassen.

Grafik und Hardcopy

Den Qualitätsfanatikern werden die GEM-Image-Grafiken, die von Wordplus ausgedruckt werden, nicht die Auflösung bieten, die man von guten Hardcopies her kennt. Sie haben jedoch die Möglichkeit, innerhalb von GEM- und TEXT-VIP mit der Tastenkombination **ALT+HELP** Hardcopies auszulösen. Selbst NEC P5/6/7- oder Epson LQ1500-Drucker werden mit den mitgelieferten Hardcopy- oder den PD-Treibern beste Qualitäten erzielen.

Worksheet Range Copy Move File Print Graph Data Quit
Global, Insert, Delete, Column-Width, Erase, Titles,

	A	B	C	D	E
1		Grafische Darstellungen			
2					
3	Quartal	1	2	3	4
4					
5	KOSTEN:				
6	Miete	1950	1950	1950	1950
7	Auto	852	952	1562	852
8	Leben	4500	7000	3500	5000
9	sonst.	2500	1800	7563	500
10	------------	------------	------------	------------	------------
11	SUMME:	9802	11702	14575	8302

Abb. 1: Arbeitsblatt

	G	H	I	J
1	Name	Syntax	|	Dokumentation
2	--------	----------------	|	------------------------------
3	\G	/GRG	|	Befehlsleiste,Graph,Reset,Graph
4		/GTS	|	Befehlsleiste,Graph,Type,Stacked-Bar
5		XB3.E6~	|	X-Range, B3.E3, Return
6		AB6.E6~	|	A-Range, B6.E6, Return
7		BB7.E7~	|	B-Range, B7.E7, Return
8		CB8.E8~	|	C-Range, B8.E8, Return
9		DB9.E9~	|	D-Range, B9.E9, Return
10		OTFKurs 4~	|	Options,Titels,First,Kurs 4,Return
11		OTS1988~	|	Options,Titels,Second,1988,Return
12		OTXQuartale 1988~	|	Options,Titels,X-Axis,Quartale 1988,Return
13		OTYKosten~	|	Options,Titels,Y-Axis,Kosten,Return
14		GV	|	Grid,Vertical
15		LAMiete~	|	Legend,A,Miete,Return
16		LBAuto~	|	Legend,B,Auto,Return
17		LCLeben~	|	Legend,C,Leben,Return
18		LDsonst.~	|	Legend,D,sonst.,Return
19		QVQ/XQ~	|	Quit,View,Quit,Macro-Ende mit /XQ
20				

Abb. 2: Grafikmacro

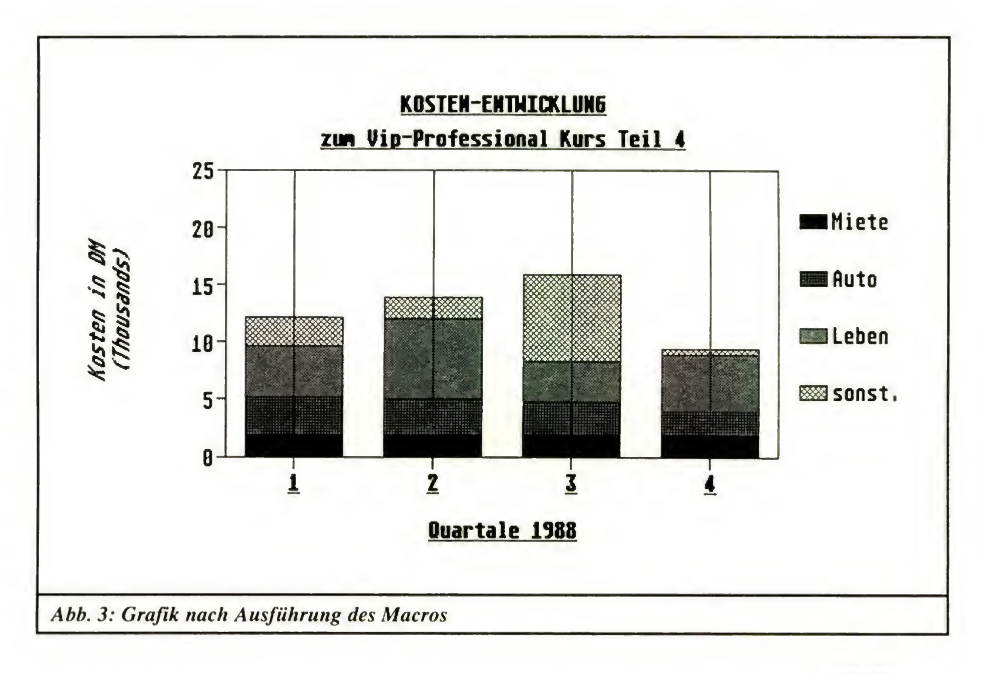


Abb. 3: Grafik nach Ausführung des Macros

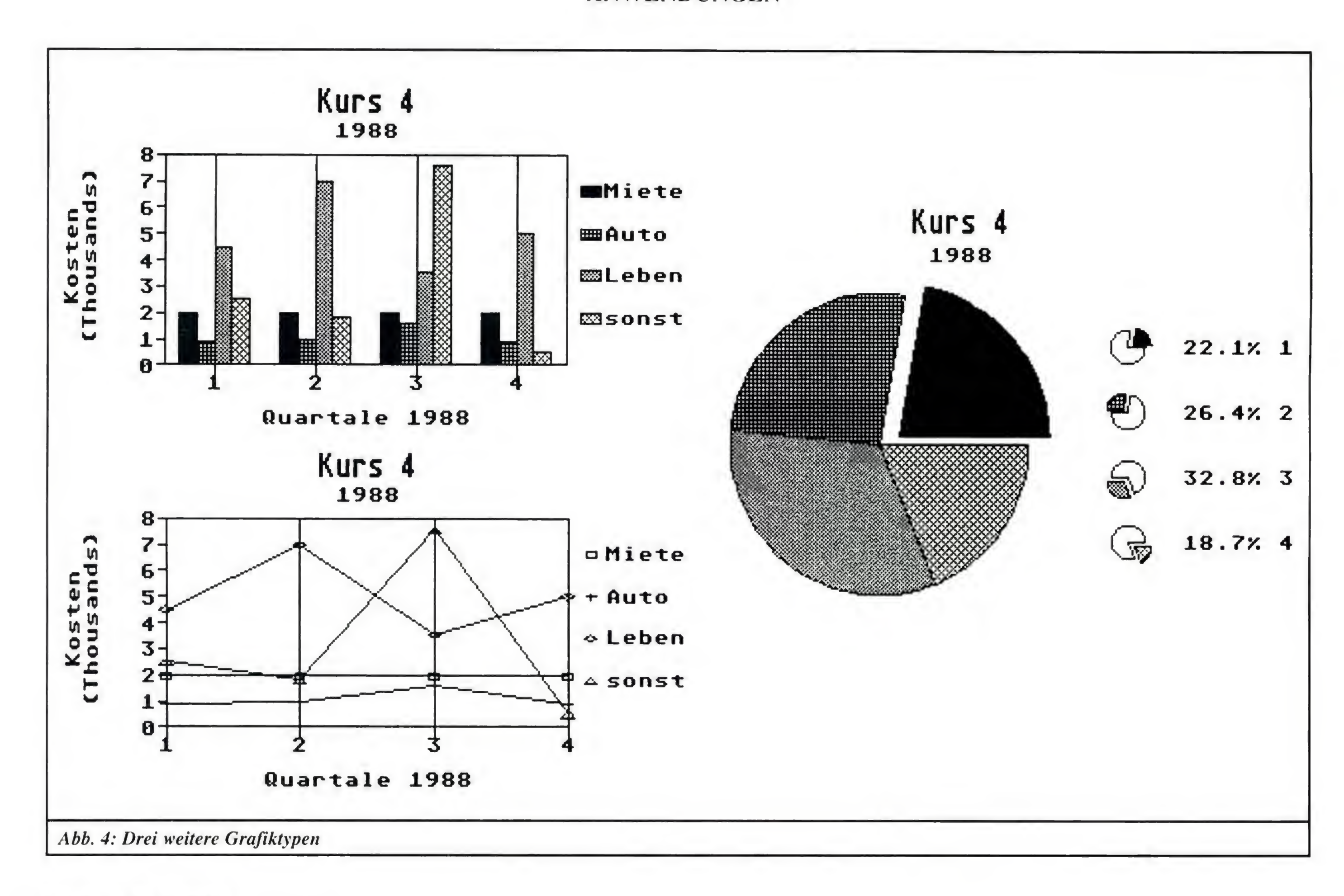


Abb. 4: Drei weitere Grafiktypen

VIP-GRAFIK und eine DIASHOW

In meinem Beruf gehört das Präsentieren von Unternehmensergebnisse zum täglichen Geschäft. Schon längst bin ich dazu übergegangen, komplexere Themenbereiche durch Grafiken aufzulockern. Solche Präsentationen werden parallel zu sogenannten "Handouts" (Dokumente) entweder über den Monitor oder wenn mehr als 3 Personen angesprochen werden sollen, über einen LCD-Overhead-Bildschirm vorgeführt.

Stellen Sie Ihre Grafiken aus VIP, Bildschirmdumps aus Tabellen, eingescannten Firmenlogos oder sogar digitalisierten Bildern (Easytizer) zu einer Slideshow zusammen. Meine Erfahrung hat gezeigt, daß einer solchen Präsentation zu folgen einen sehr hohen Akzeptanzwert hat. Wenn Sie zur grafischen Präsentation noch Fragen haben, schreiben Sie an Merlin oder direkt an mich, wir werden Ihnen gerne helfen, besser zu sein als die anderen.

Probieren, probieren,...

Spielen Sie mit den grafischen Möglichkeiten von Vip-Professional ein wenig herum. Ein kleiner Tip, ändern Sie doch mal den Grafiktype auf **LINE** oder **BAR** ab. Sieht doch auch ganz gut aus. (Abbildung 4)

Beenden wir unseren Grafikexkurs an dieser Stelle und widmen wir uns dem Pflichtteil des Kurses. Ein wenig Formeln und Funktionen möchte ich Ihnen, wie versprochen, noch beibringen.

Sonderfunktionen

Die Sonderfunktionen führen eine Vielzahl von komplexeren Aufgaben aus, wobei insbesondere auf die Möglichkeit von Tabellenverweisen hingewiesen wird.

@NA

gibt den numerischen Wert @NA zurück, wenn eine Zahl zur Vollendung einer Formel nicht verfügbar ist (not available). Diese Funktion gibt @NA in die aktuelle Zelle und in alle anderen Zellen ein, die von der Formel in dieser Zelle abhängen. @NA wird besonders beim Erstellen eines Arbeitsblattes benutzt, das Daten umfaßt, die noch nicht festgelegt wurden. Diese Funktion kann in den Zellen benutzt werden, in die diese Daten eingegeben werden müssen. Formeln, die sich auf diese Zellen beziehen, verfügen über den Wert @NA, bis die richtigen Daten angegeben werden. Da @NA in jeder Zelle steht, die von der Zelle mit @NA abhängt, wirkt sich diese @-Funktion auf das ganze Arbeitsblatt aus.

Beispiel: @IF(B1>3.2,@NA,B1) = NA, wenn der Wert von Zelle B1 größer ist als 3.2.

@ERR

diese @-Funktion wird benutzt, wenn eine Zelle den Wert @ERR haben soll. Sämtliche Zellen, die Formeln enthalten, die von dem Wert in dieser Zelle anhängen, nehmen ebenfalls den Wert @ERR an. Daraus ergibt sich eine Folgewirkung für das ganze Arbeitsblatt.

Beispiel: @IF (B1>3.2,@ERR,B1) = ERR, wenn der Wert von Zelle B1 größer ist als 3.2; ansonsten wird der Wert in B1 zurückgegeben.

@CHOOSE

Schreibweise: @CHOOSE(*SELECTORZAHL, ARGUMENT1, ARGUMENT2, ...ARGUMENTn*)

Diese Funktion benutzt die Selektorzahl, um einen Wert in der Argument-Liste: *ARGUMENT1, ARGUMENT2, ...ARGUMENTn* auszuwählen. Mit *SELEKTORZAHL* wird angegeben, welchen Wert in

der Liste @CHOOSE auswählen soll. Der Wert in der Liste lautet 0. Ist also *SELEKTORZAHL* gleich 1, so wählt @CHOOSE den zweiten Wert der Liste.

Der Benutzer muß die *SELEKTORZAHL* und mindestens einen Wert in die Argument-Liste eingeben. Bei der *SELEKTORZAHL* muß es sich um einen numerischen Wert handeln, der nicht größer sein darf als die Anzahl von Werten in der Argument-Liste.

In der Liste können beliebig viele Werte stehen, solange sie nicht 240 Zeichen pro Zelle überschreitet.

Beispiel: @ CHOOSE(1,+A1,+A2,+A3) = Wert in +A2
@ CHOOSE(EINGABE,+A1, +A2,+A3) = Wert in A2 wenn die benannte Zelle EINGABE den Wert 1 beinhaltet.
@ CHOOSE(+A1,+B1..B20)

@HLOOKUP
Schreibweise: @ HLOOKUP (*Argument, Bereich,Versatz*).

Diese Funktion führt einen horizontalen Tabellenverweis aus. Dabei wird der Wert *Argument* mit jeder Zelle in der obersten Zeile mit Eingaben im Bereich verglichen. Wird eine Zahl gefunden, deren Wert mit *Argument* übereinstimmt, so wird nicht weiter horizontal durch die oberste Zeile gegangen. Danach geht @HLOOKUP in dieser Spalte um die Anzahl von Zellen nach unten, die von Versatz angegeben wird, um die Antwort zurückzugeben. Mein Beispiel finden Sie in Abbildung 5.

@VLOOKUP
Schreibweise:@VLOOKUP (*Argument, Spaltenbereich, Versatz*)

Diese Funktion führt im Gegensatz zu @HLOOKUP einen vertikalen Tabellenverweis durch. Dabei wird der Wert *Argument* mit jeder Zelle in der ersten Spalte mit Eingaben in Spaltenbereich verglichen. Wird eine Zelle mit einem Wert gefunden, der ebenso groß ist wie *Argument*, stoppt die vertikale Bewegung in der ersten Spalte. An dieser Stelle geht @VLOOKUP um die Anzahl von Spalten weiter, die mit *Versatz* angegeben wird und ermittelt das Ergebnis.

Logische Funktionen

Die logischen @-Funktionen von VIP-Professional erzeugen Werte, die auf Er-

UNTITLED

	A	B	C	D
1		AUSGABENS	TRUKTUR	
2		1988	1989	1990
3	Leben	25000	28000	30000
4	KFZ	5000	6000	8000
5	Bildung	8000	10000	10000
6	Haus	35000	35000	35000
7				
8	@HLOOKUP(1990,A2..D6,4)	35000		

Abb. 5: @HLOOKUP

gebnisse bedingter Aussagen beruhen.

Eine bedingte Aussage wertet eine Bedingung in Form einer Gleichung aus. Die Bedingung ist entweder wahr oder falsch.

@FALSE
gibt den logischen Wert 0 zurück und kann mit anderen @-Funktionen benutzt werden, um einfach zu lesende logische arithmetische Formeln zu erstellen.

Beispiel: +A1=9
@IF(+A1=10,@TRUE,@FALSE) = 0, wenn der Wert in der Zelle A1 nicht den Wert 10 hat.

@TRUE
gibt den logischen Wert 1 zurück und kann ebenfalls mit anderen @-Funktionen benutzt werden. Ein Beispiel dürfte sich erübrigen.

@ISNA
Schreibweise: @ISNA(x)
Testet, ob x den Wert @NA enthält. (@NA steht für "NICHT verfügbar") @ISNA gibt den Wert 1 zurück, wenn x den Wert @NA aufweist; ansonsten wird 0 zurückgegeben. Diese Funktion stoppt die Folgewirkung der Werte @ERR und @NA.

@ISERR
Schreibweise: @ISERR(x)
testet, ob das Argument den Wert @ERR enthält. @ISERR gibt 1 zurück,wenn x den Wert @ERR aufweist; ansonsten wird 0 zurückgegeben. Diese Funktion stoppt die Folgewirkung der Werte @ERR und @NA.

@IF
Schreibweise: @IF(*Bedingung,x,y*)
@IF gibt den Wert x zurück, wenn die *Bedingung* **wahr** ist, oder den Wert y, wenn die *Bedingung* **falsch** ist.

Bei der *Bedingung* muß es sich um einen numerischen Wert oder eine beliebige Formel handeln, die einen numerischen Wert erzeugt. Bei x und y handelt es sich um numerische Werte.

Beispiel: @IF(9>6,D1,E1) = Wert aus D1
@IF(B3-C3>=0,B3-C3,0) = B3-C3, wenn B3-C3 positiv ist.

Statistische Funktionen

Mit den statistischen Funktionen von VIP-Professional werden Berechnungen mit Wertelisten ausgeführt. Bei numerischen Argumenten kann es sich um Zahlen,Zelladressen,Zellnamen,Bereichsadressen und Bereichsnamen handeln. Eine Liste enthält eines oder mehrere Argumente. Jedes Argument in einer Liste kann einen einzelnen Wert oder einen Bereich darstellen. Eine Liste kann einzelne Werte und Bereiche umfassen. Zum Beispiel enthält @COUNT(A1..A99,B1.B99,K8) eine gültige Argument-Liste.

VIP-Professional ignoriert leere Zellen in mehrere Zellen umfassenden Bereichen. Wird beispielsweise @AVG benutzt, um den Durchschnitt der Werte in einem zehn Zellen umfassenden Bereich zu ermitteln, und ist in diesem Bereich eine leere Zelle vorhanden, so dividiert VIP die Summe durch 9, um den richtigen Durchschnitt zu ermitteln. VIP betrachtet eine leere Zelle, die als ein Argument in einer Liste benutzt wird, als Zelle mit dem Wert 0.

@COUNT
Schreibweise: @COUNT(*Argument-Liste*)

Diese Funktion wählt die Anzahl von Zellen in der *Argument-Liste*.@COUNT ist eine der @-Funktionen, die die Folgewirkung der Werte @NA und @ERR stoppen, die in dem Abschnitt der logischen @Funktionen erläutert wurde. Enthält die *Argument-Liste* nur leere Bereiche,so lautet das Ergebnis 0. @COUNT (A1..A4)= einen Wert zwi-

UNTITLED

	A	B	C	D
1			Einnahmen 1988	
2				
3	1.Quartal	DM	150252.00	
4	2.Quartal	DM		
5	3.Quartal	DM	52425.00	
6	4.Quartal	DM	85858.00	
7				
8				
9	UMSATZQUARTALE		3	@COUNT(C3..C6)
10	SUMME JAHR	DM	288535.00	@SUM(C3..C6)
11	MINIMUM	DM	52425.00	@MIN(C3..C6)
12	MAXIMUM	DM	150252.00	@MAX(C3..C6)
13	DURCHSCHNITT	DM	96178.33	@AVG(C3..C6)

Abb. 6: Statistische Funktionen

schen 0 und 4. Ist jede Zelle in dem Bereich A1...A4 leer, so gibt @COUNT den Wert 0 zurück. Enthält der Bereich A1...A4 eine gefüllte Zelle, so gibt diese Funktion den Wert 1 zurück.

@SUM
Schreibweise: @SUM(*Argument-Liste*)
Addiert die Werte der *Argument-Liste*.

Beispiel: @SUM (A1..A99)
@SUM (EINNAHMEN,DIVI DENDEN, SONSTIGE)
@SUM(A1,A8,B9,AA99)

@AVG
Schreibweise: @AVG(*Argument-Liste*)
Diese Funktion berechnet den Mittelwert sämtlicher Werte in der *Argument-Liste*. @AVG ist gleichbedeutend mit der Formel:@SUM(*Argument-Liste*)/ @COUNT(*Argument-Liste*)

@MIN
gibt in gleicher Schreibweise wie bei @AVG den kleinsten Wert der *Argument-Liste* zurück.

@MAX
gibt den größten Wert der *Argument-Liste* zurück.

Datumsfunktionen

Mit den Datumsfunktionen werden Seriennummern zur Darstellung von Datums- und Zeitangaben generiert und benutzt. Aufgrund dessen können Datums- und Zeitangaben in Tabellenberechnungen benutzt werden. Jedes Datum zwischen dem 1.Januar 1900 und dem 31.Dezember 2099 verfügt über eine ganzzahlige Seriennummer, die als Datumsseriennummer bezeichnet wird. Die erste Datumsseriennummer ist 1 und die letzte 73050. Der 1.Januar 1900 entspricht demWert 1 und der 31.Dezember 2099 dem Wert 73050.

@TODAY
Schreibweise:@TODAY(*Jahr,Monat,Tag*)

Diese Funktion gibt die Datumsseriennummer von *Jahr, Monat und Tag* zurück. Bei *Jahr* muß es sich um eine Zahl zwischen 0 (1900) und 199 (2099) einschließlich handeln. *Monat* muß eine Zahl zwischen 1 und einschließlich 12 sein. *Tag* muß eine Zahl zwischen 1 und einschließlich 31 sein. Hier muß es sich jedoch um ein gültiges Datum für den jeweiligen Monat handeln. So kann beispielsweise 31 für Februar nicht benutzt werden. Werden für *Jahr,Monat und Tag* keine gültigen Zahlen angegeben, gibt Vip den Wert @ERR zurück.

@DAY
Schreibweise:@DAY(*Datumsseriennummer*)gibt den Tag des Monats (1-31) einer bestimmten *Datumsseriennummer* zurück.

Beispiel: @DAY(@TODAY) = Tageszahl des Systemdatums
@DAY(C6) = Tageszahl der Datumsseriennummer in Zelle C6.

@MONTH
Die Schreibweise und Ausführung ist mit @DAY zu vergleichen, nur wird eben der gültige Monat ausgegeben.

@YEAR
Die Schreibweise und Ausführung ist mit @DAY zu vergleichen, nur wird das gültige Jahr ausgegeben.

Bei der Verwendung von Datumsfunktionen in Tabellen ist auf das richtige Formatieren der Zellen zu achten. Die Datumsfunktion @TODAY erfordert beispielsweise eine Zellenbreite (COLUMNWIDTH) von >10 Stellen. Dieses Zelle muP zudem noch über die Sequenz: /,Range, **Format, Date, 1,~** formatiert werden.

Jetzt reicht es aber...!

Na ja, ich habe etwas schneller gemacht, um Sie an Ihren Urlaubsvorbereitungen nicht zu hindern, vergessen Sie nicht, das Netzkabel Ihres Ataris in den Koffer zu packen und bedenken Sie, daß Floppylaufwerke und Harddisks kein Salzwasser vertragen.

Aua, das tat weh! Meine Frau hat von meinen Atari-in-Urlaub-mitnehm-Absichten gehört und mir doch tatsächlich meine Maus geklaut.

Doch Spaß beiseite, ich möchte noch auf einige Probleme eingehen, die von vielen eifrigen Briefschreibern angesprochen wurden.

VIP von Atari auf Lotus MS-DOS

Wie bereits mehrmals ausgeführt, kann man Arbeitsblätter die auf VIP-Professional erstellt wurden, auch auf einem MS-DOS Rechner weiterbearbeiten, der Weg zurück geht ebenfalls.

Als Grundvoraussetzung sollten Sie im Besitz eines 5 1/4"-Laufwerkes sein. Dieses Laufwerk muß eine Umschaltung von 40 auf 80 Spuren (TRACKS) besitzen. Mit den 40 Spuren lassen sich MS-DOS-Disketten mit 360Kbyte formatieren, vorausgesetzt, Sie haben ein entsprechendes Formatierprogramm. (P.S.:das Buch **"Scheibenkleister"** von CB und Partner hilft Ihnen da bestimmt weiter.}

Die Steprate des Laufwerks muß softwareseitig auf 6ms eingestellt werden und los geht's. Kopieren Sie einfach Ihre fertige Anwendung, als VIP-Profi dürfen Sie ja jetzt Template dazu sagen, einfach auf die MS-DOS-Diskette. Dieses Template läßt sich ohne Probleme in jedeVersionsausgabe von Lotus 1-2-3 einladen, weiterverarbeiten und kopieren. Sollten Sie jedoch Veränderungen vornehmen, diese abspeichern und auf Ihren Atari zurückspielen wollen, muß die Translate-Routine (Übersetzung) von Lotus 1-2-3

durchlaufen werden, um die Syntax und Zeichenkompatibilität rückgängig zu machen.
Verwenden Sie dazu nur die Zielversion Lotus 1A, und alles funktioniert als wären Atari und MS-DOS Brüderchen und Schwesterlein. Zu bedenken ist bei den unterschiedlichen Lotus-Versionen nur der erweiterte Befehlsumfang, Lotus 2.0-Zusatzbefehle laufen auf der 1A- bzw. VIP-Version selbstverständlich nicht.

Der Bildschirm spielt nicht mit!

Von mehreren Anwendern habe ich gehört, daß bei einem Datenaustausch zwischen ATARI und MS- DOS- Rechner ein fehlerhafter Bildschirmaufbau beim Rückladen von Templates entstehen kann. Die Befehlsleiste ist dann nicht mehr an ihrem üblichen Platz. Abhilfe schafft hier folgende Regel:

1.) Speichern Sie Ihr Arbeitsblatt immer von Cursorposition A1 ab.
2.) Beim Rückladen und fehlerhaften Aufbau des Bildschirms folgende Sequenz eingeben: **{GOTO}A2~/WTH/WTC**

Das war es schon.

Keine Fileselektorbox!

Die meisten Kritiker an VIP-Professional vermissen eine elegante Fileselektorbox, die eben einfach zum GEM-Desktop gehört. Doch wie läßt sich dieses Manko einfach und effizient lösen? Erinnern wir uns noch einmal an den Autostart-Macro mit dem Namen **\0**. Er hat innerhalb eines Templates für die automatische Ausführung einer Befehlsfolge gesorgt. Vip-Professional stellt auf der File - Ebene ebenfalls einen solchen AUTO-Prozeß zurVerfügung. Das File **AUTO-VIP.WKS** oder **AUTO123.WKS** wird nach dem Start von VIP automatisch geladen. Wir werden das Fileselektor-Problem über dieses Auto-FILE lösen.

Richten Sie Ihre Anwenderkataloge (Ordner) nach meinem Vorschlag ein:

1.) Legen Sie einen Ordner **USER** an; öffnen Sie diesen.

2.) Sie sind im Ordner **USER** und legen folgende Unterordner an: **DIVERS, BERUF, PRIVAT, TEST**

3.) Starten Sie nun VIP und geben das AUTO-MACRO gemäß dem Listing am Ende dieses Kursteils ein

4.) Speichern Sie dieses Macro unter dem Namen **AUTOVIP.**

5.) Stellen Sie imVIP gemäß nachfolgender Sequenz die Start-Directory ein: **/WGDDA :\ USER\~U** (A:\USER steht für Laufwerk A Verzeichnis USER).

6.) Verlassen Sie nun VIP über **/QY** und kopieren Sie das File **AUTO-VIP.WKS** in das Verzeichnis **USER** sowie auch in jedes Unterverzeichnis.

7.) Bei einem Start von VIP erhalten Sie somit eine einfaches Fileselektor-Template. Um in ein anderes Unterverzeichnis zu kommen laden Sie einfach wieder **AUTO-VIP.WKS** zurück und verzweigen Sie erneut.

Gaudemium laboris acti

oder: Nach getaner Arbeit ist gut ruh'n! Das wünsche ich allen Urlaubern, Balkoniern, Hawaianern und allen, die sich ab jetzt auch noch die Tastatur und den Kopf mit Vip-Grafik zerbrechen. Wenn Sie trotzdem am heißen Meeresstrand des Atlantiks oder in einem Orangenhain in Casablanca an Vip-Probleme denken, schreiben Sie an Merlin oder direkt an mich.

Das nächste Mal beginnen wir mit der Datenbank und der weiteren Verbindung zur Außenwelt. *HDS*

Heinz D. Schultz, Ingersheimer Strasse 12
D-7140 Ludwigsburg, Tel.: 07141/38964

ENDE

```
A1:  '  B2:  '|
C2:  \-  D2:  \-  E2:  \-  F2:  \-  G2:  \-  H2:  '|
B3:  '|
H3:  '|
B4:  '|
D4:  'Mit diesem Macro koennen einzelne Anwendungen  H4:'|
B5:  '|
D5:  'in den Unter-Katalogen angewaehlt werden.
H5:  '|
B6:  '|
H6:  '|
B7:  '|
D7:  'Positioniere den Cursor auf die Anwendung  H7:  '|
B8:  '|
D8:  'und druecke RETURN
H8:  '|
B9:  '|
H9:  '|
B10:  '|
D10:  '( Mit ALTERNATE + M kommen wir ins Menue )
H10:  '|
B11:  '|
H11:  '|
B12:  '|
D12:  'Heute ist der :
F12:  (D1) @TODAY
H12:  '|
B13:  '|
H13:  '|
B14:  '|
D14:  'Viel Spass bei der Arbeit.   (c)HDS 1999
H14:  '|
B15:  '|
H15:  '|
B16:  '|
```

```
H16:  '|
B17:  '|
C17:  \-  D17:  \-  E17:  \-  F17:  \-  G17:  \-  H17:  '|
A24:  '\M
C24:  '/xg\0~
A26:  '\0
C26:  '/wtc{CALC}{HOME}/wgpe/xgMENU~
A28:  'MENU
C28:  '/xmc29~
C29:  'Divers
D29:  'Beruf
E29:  'Privat
F29:  'Test
G29:  'A-Drive
H29:  'B-Drive
I29:  'C-Harddisk
J29:  'Quit
C30:  'Diverse Anwendungen
D30:  'Geschaeftliche Notwendigkeiten
E30:  'Private Budget-Verwaltung
F30:  'Testdatei fuer Templates
G30:  'auf Laufwerk A zugreifen
H30:  'auf Laufwerk B zugreifen
I30:  'auf Harddisk C zugreifen
J30:  'Leeres Template erzeugen
C31:  '/fdA:\USER\Divers~/fr
D31:  '/fdA:\USER\Beruf~/fr
E31:  '/fdA:\USER\Privat~/fr
F31:  '/fdA:\USER\TEST~/fr
G31:  '/fdA:\~/fr
H31:  '/fdB:\~/fr
I31:  '/fdC:\~/fr
J31:  '/wey~/xq~
```

Das Fileselektor-Listing



WIE BITTE ?

Eine ausbaufäh. FAKTURIERUNG mit integrierter ADRESSEN- und LAGERVERWALTUNG

- die auch mit 1-s. Floppy, 512 kB u. Monochrommonitor läuft ?
- viele Möglichkeiten u. persönliche Einstellungen erlaubt ?
- dank durchdachter Menueführung unglaubl. schnell, bequem u. leicht erlernbar ist ?
- und die trotzdem nur 99,- kostet ?

JA, PegaFAKT

Kostenloses Info / Händleranfr. erw.





Software

TEAMBASE .... **DM 375.-**

Eine neue Generation von Datenbank-Programm

- Beliebig viele Datensätze
- Als Programm und Accessory benutzbar
- Schnittstelle zu 1ST Word für Serienbriefe.
- Daten Import und Export
- Minimaler Speicherbedarf durch Datenkompression
- Datenschutz durch Verschlüsselung und/oder Passwortvergabe.
- Hoher Bedienungskomfort durch volle GEM-Nutzung, gleichzeitig weitgehend über die Tastatur-steuerbar.
- Integrierte Kommandosprache.
- vieles mehr (auch Kundenapplikationen)

Kommandozeileninterpreter ST - DCL ... **DM 298,-**

ST - DCL erlaubt es, auf einem Atari ST unter einer Kommandoumgebung zu arbeiten,die der VAX/VMS-Digital Command Language (DCL) nachempfunden ist.

Arzt-Software : Aus der Praxis für die Praxis

Privatliquidation .... **DM 375,-**
Praxisbuchhaltung .... **DM 375,-**

Sämtliche Standart-Software lieferbar.

z.B. Application Systems, C.A.$.H., CCD, DMC, Drews EDV + Btx, G DATA, GFA, GST, HAGERA, Hyper-Soft, Kniss-Soft, Kieckbusch, Prodata, Technobox, u.v.m.

Hardware

Atari 1040 STF (komplett) .... **DM 1498,-**
Atari Mega ST 2/4 .... **auf Anfrage**
star LC 10 .... **DM 698,-**
star LC 24-10 .... **DM 1098,-**
NEC P6 plus .... **DM 1698,-**
NEC P 2200 .... **DM 998,-**

Alle Drucker mit deutschem Handbuch und Kabel

vortex HD plus 20 .... **DM 1149,-**
vortex HD plus 30 .... **DM 1398,-**
vortex HD plus 40 .... **DM 1998,-**
vortex HD plus 60 .... **DM 2198,-**

vortex Test-Wochen vom 1.8.bis 31.8.

...TESTEN, VERGLEICHEN, GEWINNEN!

LOGITEAM

Kölner Str. 132 • 5210 Troisdorf • Tel.02241/71897-98

MÖCHTEN SIE BEI UNS SCHREIBEN?

Liebe Leser, die ST-COMPUTER gibt es nun schon fast drei Jahre. Die ST-COMPUTER machte sich als erste monatliche ST-Zeitschrift Deutschlands daran, Licht in das Dunkel um das damals noch unbekannte Wesen ATARI ST zu bringen. Viele Geheimnisse dieses Rechners wurden gelüftet und in Form von Programmen oder Artikeln jedem zugänglich gemacht.

Doch nicht alle Gebiete lassen sich von uns abdecken. Der einzelne Leser ist hierbei gefordert, seine eigenen Erfahrungen und Entdeckungen zu präsentieren. Deshalb der Aufruf an alle Leser:

Beteiligen Sie sich aktiv an der Gestaltung der ST-COMPUTER.

Wir möchten Ihnen auf diesen Seiten die einzelnen ST-COMPUTER-Rubriken darlegen und auf einige Besonderheiten verweisen, die Sie wissen sollten, wenn Sie diesem Aufruf folgen.

PROGRAMMIER PRAXIS

Diese Rubrik ist zur Zeit die beliebteste Sparte unserer Leser. Hier versuchen wir jeden Monat interessante Listings, die ein wenig Licht ins Dunkel um die Programmierkunst des ATARI ST bringen, vorzustellen. Daß hierbei eine aktive Mitarbeit unserer Leserschaft erforderlich ist, können Sie sich sicherlich denken.

Was sollten Sie beachten?
Die PROGRAMMIER PRAXIS enthält Programme oder Routinen zu allen Bereichen der ST-Programmierung. Die Sprache überlassen wir dabei Ihnen. Wichtig ist nur, daß die Listings entweder nützlich oder lehrreich sind.

- Schicken Sie uns den Source-Code und das startbare Programm auf Diskette zu.
- Der Artikel sollte sich ebenfalls auf Diskette befinden. Bei vorhandenem Drucker bitten wir Sie, einen Ausdruck von Text und Listing beizulegen (wenn möglich, in 1st_Word-Format).
- Falls der Artikel Bildschirmfotos benötigt, so sollten Sie die Bilder (als Screendump) auf Diskette abspeichern. Hier sind die Formate beliebig (Doodle, Degas usw.).
- Die Länge des Listings sollte dem Nutzen des Programms entsprechen, d.h., daß man auf Schnörkel, wenn diese das Programm unnötig verlängern, besser verzichtet.
- Wir haben diese Rubrik sinnvoll erweitert. Neben einer Beschreibung der allgemeinen Funktionsweise des Programms (oder der Routine) sollte jetzt ein Struktogramm die Funktionsweise noch besser darstellen. Es soll zusätzlich die Möglichkeit geschaffen werden, daß auch andere, die diese bestimmte Programmiersprache nicht beherschen, Ihr Programm (Routine) in eine andere Sprache umsetzen können. Dieses Struktogramm kann mit jedem gängigen Zeichenprogramm erstellt werden und ebenso auf Diskette eingesandt werden.

Besitzen Sie kein Malprogramm, so schicken Sie uns einfach eine Handskizze, die dann von einem Grafiker hier in der Redaktion "druckreif" gezeichnet wird.

- Bei der Erstellung Ihres Listings beachten Sie bitte,daß es nach Möglichkeit nicht breiter als 55 Zeichen sein sollte.
- Ein Beitrag in der Programmier Praxis muß nicht unbedingt aus einem ablauffertigen Programm bestehen. Auch kurze Routinen, die der Anwender in sein eigenes Programm einbinden kann, sind gefragt. Auf diese Art und Weise lassen sich leicht leistungsstarke Bibliotheken erstellen.

Ihrer Phantasie wird dabei freier Lauf gelassen.
Kennwort: PROGRAMMIER PRAXIS

PROJEKT

Unter einem Projekt verstehen wir längere Programme (siehe Extended VT52-Emulator oder der neue Linker für das GST-Format), die in einem oder mehreren Heften abgedruckt werden können. Wir benötigen allerdings das ganze Projekt an einem Stück, d.h komplett. Nur dann können wir entscheiden, ob es korrekturbedürfig ist oder noch Erweiterungen nötig sind. Für die Sparte Projekt gilt im allgemeinen das gleiche wie für die PROGRAMMIER PRAXIS. Denken Sie immer daran, daß ein Listing schon mal etwas länger werden kann, aber noch abtippbar und dementsprechend nützlich sein sollte. Den Maßstab für die Länge des Programms können Sie leicht selbst beurteilen. Wieviele Zeilen würden Sie noch abtippen?
Kennwort: Projekt

HARDWARE

Hardware-Projekte sind in der ST-COMPUTER keine Seltenheit. Wir haben einen Digitizer vorgestellt und erklärt, aus dem Easyprommer haben wir einen weiteren leistungsfähigen Eprommer gebaut, nämlich den Junior Prommer. Unser Umbau des ATARI-Monitors hat eine große Resonanz gehabt und der Scanner war eine preisgünstige Alternative für alle, die mit dem ATARI ST viel im graphischen Bereich anfangen möchten.
Es müssen aber nicht unbedingt sehr große Schaltungen sein. Eine kleine, aber sinnvolle Erweiterung oder ein Zusatzgerät, oder...

Was müssen Sie beachten?
Zu einem Hardware-Projekt gehört außer der Dokumentation (die die Funktionsweise des Gerätes sowie dessen Bedienung erklären sollte) ein Schaltplan (kann auch mit einem Malprogramm erstellt oder als Handskizze beigelegt werden), eine Stückliste und ein Platinenlayout. Bei größeren Projekten muß unbedingt ein lauffähiger Prototyp mitgeliefert werden. Dieser wird nach Begutachtung umgehend zurückgeschickt. Hat das Projekt auch noch einen Softwareteil, so sollte dieser nach den oben genannten Bedingungen (Programmier Praxis sowie Projekt) auf Diskette mitgeliefert werden.
Kennwort: 'Hardware'

GRUNDLAGEN

Schwerwiegende Themen haben wir bis jetzt in dieser Sparte behandelt. So konnte man in den Floppyspielereien in mehreren Folgen das Innenleben einer "Diskette" verfolgen. Die internen Geheimisse des GEMDOS begleiten Sie zur Zeit immer noch in unserer Serie "Auf der Schwelle zum Licht".
Für gute Grundlagenartikel über den ST und die Welt sind wir jederzeit empfänglich. Themen gibt es genügend. Haben Sie sich in einem bestimmten Bereich erfolgreich hineingearbeitet und wissen darüber zu berichten, so fassen Sie Ihre Erkenntnisse doch einfach in einem Artikel zusammen und schicken ihn uns zu.

Was Sie beachten sollten!
Hier gelten ebenfalls die Regeln, die wir oben genannt haben. Also Text auf Diskette (in 1st_Word-Format), keine Trennung, linksbündig, kein Blocksatz, Bilder auf Diskette.
Kennwort: "GRUNDLAGEN"

ANWENDUNGEN

Unter diesem Motto möchten wir eine Reihe von praktischen Anwendungen auf dem ATARI ST vorstellen. Den Anfang machte damals der Artikel JuriSTische Anwendungen. Dazu kam die VIP-Serie und "Vom Papier zur Datenbank". Damit dies nicht der letzte Artikel dieser Art bleibt, möchten wir Sie hiermit dazu aufrufen, uns Ihre Erfahrungen und Anwendungen mit dem ATARI ST im alltäglichen Leben mitzuteilen. Bitte schicken Sie uns Ihre Berichte und eventuelle Bilder, die im Beitrag vorkommen auf Diskette.
Kennwort: "ANWENDUNGEN"

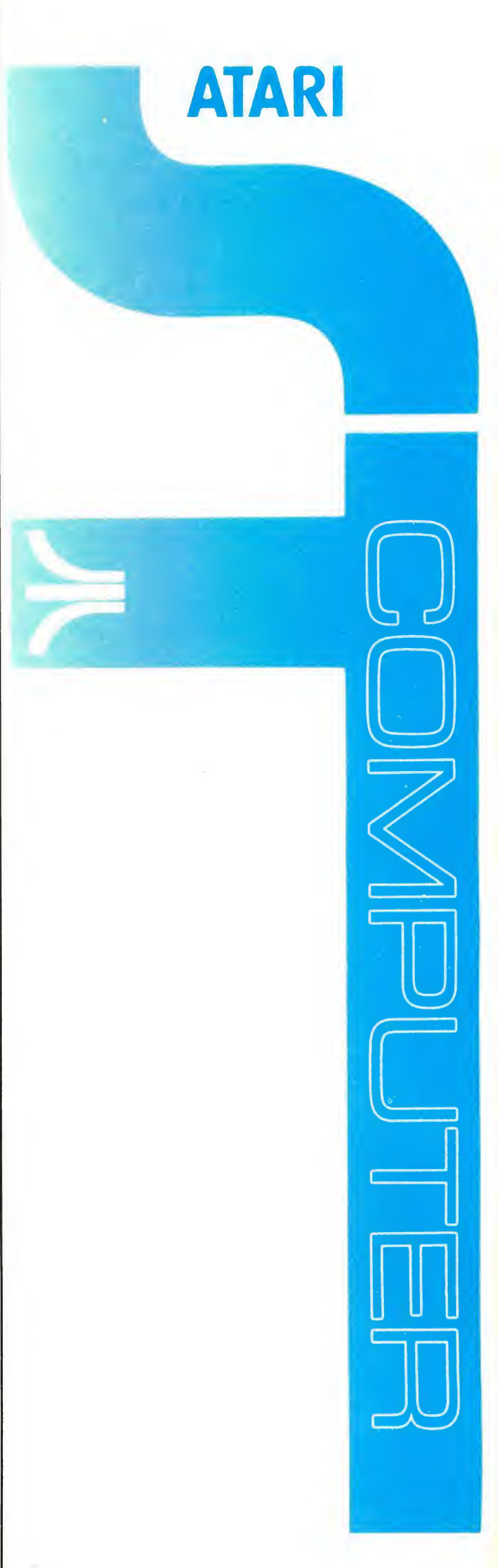
ST-ECKE

Hier wird ein wenig auf die softwaremäßigen Innereien des Rechners eingegangen. So haben wir zum Beispiel die lange undokumentierten AES-Routinen Appl_tplay und Appl_trecord erklärt. Hier sind also viele wichtige Tips zum Betriebssystem zusammengefaßt. Ansonsten gelten die gleichen Einsendebedingungen wie bei der PROGRAMMIER PRAXIS.

Wir hoffen, daß Sie sich mit dem einen oder anderen Artikeln an der Gestaltung der ST-COMPUTER beteiligen werden. Natürlich wird Ihre Mühe auch honoriert. Ihre gesamten Unterlagen werden zurückgeschickt und die Diskette zurückerstattet. Bei Fragen wenden Sie sich bitte direkt an die Redaktion und fragen nach Herrn Merino, der Ihnen gerne weitere Auskünfte gibt.

Do 14-17 Uhr Tel. 06196/481811

Kontaktadresse:
'Merlin'-Computer GmbH
ST-COMPUTER Redaktion
Z.Hd. Herrn Merino
Kennwort: ____________
Industriestr. 26
6236 Eschborn



RELAX

Hallo Leute !

Herzlichen Dank für die vielen Zuschriften auf den Wettbewerb in Ausgabe 6/88. Sie erinnern sich vielleicht noch: 15 Computerspiele sollten unter allen Einsendern verlost werden. Sie brauchten nur 2 Lieblingsspiele auf eine Postkarte zu schreiben und sie uns zu schicken. Aufgrund der großen Resonanz soll die Leserhitparade eine ständige Einrichtung bleiben. Ich wünsche Ihnen, wie immer, viel Spaß beim Lesen. Auf bald!

Ihr Carsten Borgmeier

News



Einen hörenswerten Sound hat "**Thundercats**" von "Elite". Eine Katze läuft bei ruckfreien, horizontalem Scrolling (es geht also doch!) über den Bildschirm und erschlägt mit einem Schwert Kobolde und Riesenvögel. Technisch ist "Thundercats" gut gelungen, spielerisch hat Elites neuestes Spiel jedoch nichts zu bieten. Preis: 65 DM

Info: Leisuresoft

Ums Erobern geht es in "**Lords of Conquest**", einem sehr zweifelhaften Strategiespiel von "Electronic Arts". Das Spiel erinnert ein bißchen an das Brettspiel "Risiko". Es gilt, durch Eroberung einzelner Länder die Weltherrschaft an sich zu reißen. Dabei kann sich der Spieler verschiedene Gebiete aussuchen, in denen er seinen strategischen Krieg durchführen möchte.

Info: Rushware



Die beiden icongeführten Adventures "**Uninvited**" und "**Shadowgate**" von "Mindscape" sind jetzt auch für den Atari ST in Monochromversion erhältlich.

Info: Profisoft



"Diamand Software" präsentiert ein neues Billardspiel mit dem Titel "**American Pool**". Der Billardtisch ist dabei aus der Vogelperspektive zu sehen.

Info: Profisoft

"Activision" hat ein neues Adventure mit digitalisierten Grafiken, einem intelligenten Parser und einer icongeführten Steuerung zu bieten. In der Verpackung von "**Mindfighter**" liegt das "Buch zum Spiel". Um den 150-seitigen Roman und das Game richtig genießen zu können, bedarf es guter Englischkenntnisse. Die Handlung des Adventures ist sehr interessant. Parapsychologiestudent Robin reist in die Zukunft und muß dort feststellen, daß die Welt von einem nuklearen Holocaust zerstört worden ist. Robin hilft den Überlebenden, eine Gemeinschaft aufzubauen und reist zurück in die Gegenwart, um dort den dritten Weltkrieg zu verhindern. Activision hat ein hochpolitisches und sozialkritisches Spiel geschaffen, das besonders für Erwachsene geeignet ist. Preis: ca 80 DM

Info: Leisuresoft, Ariolasoft

"**Liberator**" von Tynesoft ist ein Spiel für Schlichtgeister. Mit einem eigenartigen Schwebegefährt fährt man über eine Planetenoberfläche und schießt auf feindliche Aliens. Damit man lange genug ballern kann, muß man Munitionskisten einsammeln. (Gähn !)

Info: Rushware

Blamabel, was "Red Rat Software" mit "**Screaming Wings**" bietet. Bei der vertikal scrollenden Ballerei steuert der Spieler ein Flugzeug und muß andere Flugzeuge vom Himmel holen. Die Grafik ist in schlichten Grüntönen vermischt mit einigen blauen Pfützen gehalten. Wenn sich mehr als vier Flugzeuge auf dem Bildschirm befinden, flackert das Ganze wie ein Kaminfeuer. "Screaming Wings" muß man nicht haben!

Info: Rushware

Auch in diesem Monat gibt es wieder zwei Gauntlet-Imitationen zu vermelden. Die eine ist von "U.S Gold" und heißt "Shockled", die andere nennt sich "Pandora" und stammt von "Firebird".

Info: Leisuresoft

20000 Meilen unter dem Meer



Die utopischen Abenteuerromane des 1905 verstorbenen französischen Schriftstellers Jules Verne scheinen die Softwareautoren zu faszinieren. Nach "Reise um die Welt in 80 Tagen" (Rainbow Arts) und "Reise zum Mittelpunkt der Erde" (Chip Software) liegt mit "20000 Meilen unter dem Meer" von Coktel Vision schon die dritte Versoftung eines Jules Verne-Romans vor.

Während es sich bei den ersten beiden Jules Verne-Programmen um reine Actionspiele handelt, hat das französische Softwarehaus Coktel Vision aus dem Romanstoff ein Adventure fabriziert. Das Computerspiel hält sich weitgehend an die Romanhandlung. Um sich über die Hintergrundstory zu informieren, braucht man nicht eine umfangreiche Anleitung zu wälzen. Sie wird im Vorspann durch animierte Bilder anschaulich gemacht. Zuerst erscheint eine Zeitung auf dem Bildschirm. In der "Coktel Post" ist zu lesen, daß das seit Wochen gefürchtete Seeungeheuer wieder ein Schiff in Tausend Stücke geschlagen hat.

Ein Expeditionsschiff mit Namen "Abraham Lincoln" läuft aus, um das schreckliche Seemonster zu fangen. An Bord befindet sich der berühmte Professor Arronnax, der wissenschaftliches Interesse an dem Monster zeigt. Nach einer Kollision auf hoher See, findet er sich an Bord des Unterseebootes Nautilus. Er ist völlig verwirrt, da es Unterseeboote zu seiner Zeit eigentlich gar nicht gab. Als Gefangener Kapitän Nemos lernt Arronnax eine wunderbare Welt kennen, nämlich die Unterwasserwelt. So beginnt ein faszinierendes Adventure mit eindrucksvollen Grafiken und einer intelligenten Steuerung.

Texteingaben sind in Coktel Visions neuestem Abenteuerspiel überflüssig. Auf dem Bildschirm sieht man eine Computergrafik, die Ähnlichkeit mit einem Gemälde hat. Beim ersten Bild handelt es sich um das Arbeitszimmer von Kapitän Nemo. Mit einer Hand, die per Maus über den Bildschirm gesteuert wird, kann man Gegenstände oder Personen anklicken. Bewegt man die Maus beispielsweise auf eine zusammengerollte Karte, die auf Nemos Schreibtisch liegt, kann man die Karte lesen. Klickt man eine der Personen an, kann man in einem Textfenster lesen, was die Charaktere zu sagen haben. In den verschiedenen Räumen des Schiffes wie der Bibliothek, Nemos Arbeitszimmer, Aronnax Kabine, der Brücke, dem Kommandoraum, aber auch auf einer einsamen Insel und im tiefen Ozean spielt sich die Handlung ab. Ziel des Spielers ist es, soviel wie möglich über Nemos wundersames U-Boot und die Unterwasserwelt zu erfahren.

Alle Erkenntnisse müssen der Nachwelt im Tagebuch des Professors erhalten werden. Wenn man genug Informationen gesammelt hat, sollte man versuchen, aus dem U-Boot zu entkommen. Im Spiel sind zwei Actionszenen integriert. In einer muß man auf einer einsamen Insel einen Schatz finden, in einer anderen kämpft der Professor unter Wasser, bewaffnet mit einer Harpune, gegen gefährliche Haifische. "20000 Meilen unter dem Meer" ist ein Adventure mit viel Liebe zum Detail. Die Grafiken sind eindrucksvoll gezeichnet und teilweise animiert. Coktel Vision hat einen eigenen Stil entwickelt, wenn es um erstklassige Adventuresoftware geht. "20000 Meilen unter dem Meer" gibt es komplett in deutscher Sprache. Prädikat: Besonders wertvoll !

CBO


20000 MEILEN UNTER DEM MEER

Kategorie: Grafikadventure
Besonderheiten: Gemäldeartige Grafiken
Spieler: 1
Monitor: color
Steuerung:
Hersteller: Coktel Vision
Info bei: Bonico
Wertung:

Fußball Flop

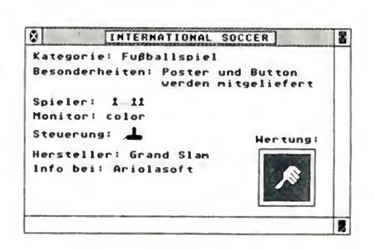


1988 ist das Jahr der Fußball-Europameisterschaft. So ist es nicht weiter verwunderlich, daß in diesem Jahr nach "International Soccer" mit "Euro Soccer 88" schon das zweite Fußballspiel für den Atari ST auf den Markt kommt. Hersteller "Grand Slam" veröffentlichte das Spiel wenige Tage vor dem Eröffnungsspiel. Ich habe mir das Programm für Sie etwas genauer angesehen: Auf den ersten flüchtigen Testerblick macht "Euro Soccer 88" noch einen guten Eindruck: Der Turniermodus der realen Kicker wurde realitätsgetreu nachempfunden. Der Spieler bestimmt die Geschicke einer von acht europäischen Mannschaften, die in zwei Gruppen jeweils gegeneinander spielen. Nach Viertelfinale und Halbfinale treten die beiden Gruppenersten im Finale gegeneinander an und spielen um den begehrten Titel. Alle Partien, in denen das Team des Spielers nicht involviert ist, werden vom Computer simuliert. Vor jeder Runde gibt der ST die Ergebnisse der anderen Mannschaften auf dem Bildschirm aus. Die Partien mit dem Team des Spielers werden auf dem Bildschirm in einem Stadion ausgetragen. Horizontales Scrolling, schön große Spielersprites in einer hübsch colorierten und animierten Grafik entzücken das Auge. Die Tribüne im Hintergrund gefällt ebenfalls durch eine detaillierte Grafik. "Ein schönes Fußballspiel" dachte ich, als der Schiedsrichter das Match anpfiff.

Doch meiner anfänglichen

Verzückung sollte ein schnelles Ende bereitet werden. Spielerisch ist "Euro Soccer 88" nämlich eine Katastrophe. Wie ich zu dieser Erkenntnis komme, können Sie in den nächsten Zeilen lesen: Bei einer Europameisterschaft sollte man davon ausgehen können, daß die teilnehmenden Mannschaften über ein gewisses Maß an Spielstärke verfügen. Dies ist allerdings nicht bei Grand Slams Fußballspiel der Fall. Schon nach wenigen Trainingsspielen kann man ohne große Schwierigkeiten Tuniersieger werden. Es wurde schnell deutlich, daß es mit der Spielstärke des Computers nicht weit her ist. Noch schwächer als die Spielstärke wurde die Steuerung gelöst. Über dem Spieler, der am nächsten zum Ball steht, schwebt ein Pfeil. Nur dieser Kicker kann vom Spieler gesteuert werden. Wenn man einen Teamkameraden einen Paß zuspielt, müßte der Pfeil eigentlich zum dem Mannschaftskameraden, der jetzt am nächsten zum Ball steht, wechseln. Manchmal geschieht das auch, aber nicht immer. Läuft man mit dem durch einen Pfeil gekennzeichneten Spieler aus dem Bild, kann es vorkommen, daß kein anderer Spieler durch den Pfeil markiert wird. In solchen Situationen kann man nur zuschauen, was der Gegner mit dem Ball macht. Auf die Dauer ist sowas natürlich frustrierend. Ein kollektives Kicken mit Doppelpässen und Flanken ist so kaum möglich. Man muß versuchen, alleine zum Torerfolg zu kommen. "Grand Slams" Fußballspiel ist ein Flop, für den jede Mark zu schade ist. Die Tatsache, daß der Verpackung noch ein Ansteckbutton und ein Poster beiliegen, kann nicht zur Ehrenrettung beitragen. 65 DM für ein derartig schwaches Spiel ist eine Unverschämtheit. Vom Hersteller des Spiels Fred Feuerstein, Grandslam, (Test in ST Computer 6/88) habe ich mir wesentlich mehr erhofft. Na ja, vielleicht das nächste Mal!

CBO



Abenteuer in Indien

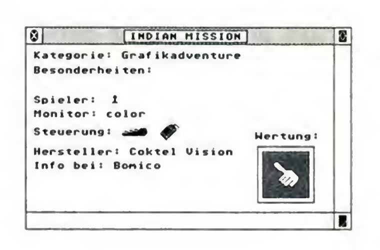


In "Indian Mission" von "Coktel Vision" sind Sie Clark Bokel, ein Professor okkulter Wissenschaften. Ihre Studienkameraden verspotten Sie aufgrund Ihres sonderbaren Berufes, aber das stört Sie wenig, denn Sie üben Ihren Beruf mit großer Leidenschaft aus. Nach einer jahrelangen Forschungsarbeit über Praktiken der indischen Zauberkunst erfahren Sie von einer geheimnisvollen Botschaft, die ein weiser Hindu vor seinem Tod in einem alten Haus in der Normandie versteckt hat. Sie beschließen, in die Normandie zu reisen und sich die Botschaft anzusehen. Das Abenteuer kann beginnen! Der Spieler steuert Bokel per Maus oder den Pfeiltasten auf der Tastatur durch die Abenteuerwelt. Man hat die Aufgabe, Bokel durch verschiedene Szenarien zu steuern. Dabei stehen Hindernisse im Weg (wilde Tiere, Schränke, Gegenstände, feindlich gesinnte Personen...), die man umgehen, beseitigen oder benutzen muß, um in der Spielhandlung zu avancieren. Ein Energiestreifen zeigt an, wie es um das Wohl Bokels bestellt ist. Ist der Streifen nicht mehr zu sehen, hat Bokel sein Leben ausgehaucht. An Lebensenergie verliert man beispielsweise durch einen Schlangenbiß. Mit vier verschiedenen Waffen kann man sich gegen wilde Tiere oder andere feindlich gesinnte Personen zur Wehr setzen.

Revolver, Schwerter, Nebelspray oder Karate sind wirksame Waffen gegen alle Widersacher. Der Energiestreifen wird größer, wenn man Früchte oder andere Nahrungsmittel einsammelt. Bokel kann während des Spielverlaufs drei besondere Fähigkeiten erwerben. So ist es möglich, sich in eine Feuerkugel zu verwandeln, um in gefährlichen Situationen besser entkommen zu können. Neben den Waffen und den drei Fähigkeiten gibt es zahlreiche Gegenstände, die für den weiteren Spielverlauf von immenser Bedeutung sind. Das Spiel beginnt vor dem Haus des weisen Inders in der Normandie. Bokel kann das Gemäuer allerdings nicht betreten, da ein bissiger Hund vor der Haustür liegt. Also muß er ein Nebelspray einsammeln und damit den Hund ausschalten. Doch selbst, als der Hund aus dem Weg geräumt ist, ist es für Bokel nicht möglich, das Haus zu betreten, weil die Haustür verschlossen ist. Ein Rabe hat den Schlüssel gestohlen. Um dem Federvieh den Schlüssel abzunehmen, muß Bokel eine Brücke passieren. Doch da lauert eine giftige Schlange. Mit einem Schuß aus der Nebelspraydose ist die Schlange eliminiert. Man nimmt jetzt den Schlüssel und geht zur Haustür, die sich nunmehr öffnen läßt. Im Haus muß man versuchen, einen alten Schrank aufzubrechen. Dort liegt eine Nachricht des weisen Inders. Er hat in geheimen Texten entdeckt, daß alte verfluchte Götter eines Tages auferstehen werden, um die Welt zu vernichten. Der Inder schreibt in der Botschaft, wer immer diese auch findet, soll aus Mitleid mit der Menschheit diese vier Gottheiten zerstören. Damit ist Bokels Aufgabe für seine "Indian Mission" klar. Auf geht's zum Flughafen. In Indien angekommen, gibt es viele Puzzles zu lösen, um schließlich die vier Götter vernichten zu können. "Indian Mission" ist ein einzigartiges Abenteuerspiel mit gut gelungener Grafik und einer interessanten Spielhandlung. Schade, daß so wenig Sound zu hören ist. Mit etwas mehr Sounduntermalung und einer präziseren Maussteuerung wäre "Indian Mission" das ultimative Abenteuerspiel.

Trotz dieser kleinen Schönheitsfehler bleibt das Spiel noch ein ausgezeichnetes Computerspiel. Man darf nur hoffen, daß Coktel Vision weiterhin so brilliante Spiele liefert.

CBO



Gauntlet II, die Fortsetzung



Darauf haben die Gauntlet-Fans lange gewartet. “Gauntlet II” ist da. Hersteller “U.S. Gold” hat einen würdigen Nachfolger zum Spieleklassiker Gauntlet kreiert. Bis zu vier Spieler können sich gleichzeitig an Gauntlet II erfreuen. Bei drei oder mehr Spielern benötigt man allerdings ein spezielles Joystickinterface, das man bei U.S Gold bestellen kann. Das Spielprinzip der oft kopierten Automatenumsetzung ist denkbar einfach.

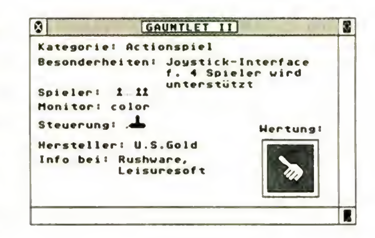
Jeder Spieler steuert seinen Charakter durch Labyrinthe und muß dabei ein Feld erreichen, auf dem “Exit” steht. Die vier Helden von Gauntlet II sind genau dieselben wie beim Vorgänger, nämlich Krieger Thor, Walküre Thyra, Kobold Questor und Hexenmeister Merlin (er hat übrigens nichts mit “Merlin Computer” zu tun, auch wenn wir manchmal hexen, um die ST-Computer rechtzeitig fertigstellen zu können).

Der Spieler steuert seinen Charakter durch ein Labyrinth und versucht, den Ausgang zu finden. Währenddessen begegnet man Monstern und Gespenstern, die an der wertvollen Lebensenergie nagen. Um neue Energie zu bekommen, braucht man nur kraftspendende Speisen einzusammeln. Neben den Speisen gibt es noch viele andere Gegenstände wie Zaubertränke, Amulette oder Schatzkisten im Labyrinth, die alle besondere Vorzüge und Nachteile für den Spieler bieten.

Insgesamt gibt es 10 verschiedene Monster, die dem Helden ans Leder wollen. Von primitiven Gespenstern bis zu Giftpfützen und steinewerfenden Kobolden ist alles vertreten. Das Geschehen präsentiert sich aus der Vogelperspektive, so daß der Bildschirm in alle acht Richtungen scrollt. Das tut er auch, wenn nur mit einem kleinen Ruck. Für ST-Verhältnisse ist das Acht-Wege Scrolling jedoch gut gelungen. Die Grafik der Charaktere und des Labyrinths ist nicht sonderlich spektakulär. Bewundernswert bei der Grafik ist eigentlich nur die Tatsache, daß unzählige Sprites gleichzeitig auf dem Bildschirm dargestellt sind. Darin liegt auch der Reiz des Spiels. Was glauben Sie, wieviel Spaß es macht, gegen 30 Monster gleichzeitig mit Joystick und Feuerknopf vorzugehen? Beim Soundtest fiel die Sprachausgabe auf, die originalgetreu vom Automaten übernommen wurde.

Fazit: Gauntlet II ist ein unterhaltsames Game, mit dem mehrere Spieler zusammen viel Spaß haben können. Der zweite Teil bietet mehr Monster, Gegenstände und Schikanen als das Vorgängerprogramm. Gauntlet II ist meiner Meinung nach noch um ein paar Klassen besser als der beliebte Vorgänger. Ich kann Ihnen diese sehr gute Automatenumsetzung nur wärmstens empfehlen.

CBO



Luxusballerspiel

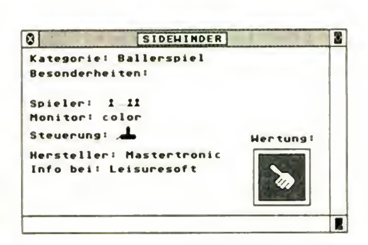


Das gesamte Universum ist mal wieder bedroht. Ein Sternenkreuzer mit dem bezeichnenden Namen Starkiller rast auf die Sonne zu, um sie zu zerstören. Sie müssen als Pilot eines ultramodernen Raumschiffes in den riesigen Sternenkreuzer fliegen und ordentlich aufräumen. Wenn Sie in seinen fünften Abschnitt des Sternenkreuzers geflogen sind, können Sie die Zentraleinheit des gigantischen Raumschiffs einäschern. Nur so kann das Unheil vereitelt werden. Eine solche Hintergrundstory nach dem Muster “Held muß Bedrohung von der Erde oder dem Universum abwehren”, ist um nahezu jedes Ballerspiel gestrickt. Bei SIDEWINDER von MASTERTRONIC ist die Hintergrundstory eigentlich überflüssig. Der Spieler ist auch ohne sie bestens motiviert. SIDEWINDER ist nämlich ein Ballerspiel mit Effekten und Gameplay vom Feinsten. Man fliegt mit seinem Raumschiff bei vertikalem Scrolling durch den Sternenkreuzer.

Sidewinder besteht aus fünf verschiedenen Levels, die alle andere Schikanen bereithalten und grafisch komplett verschieden gestaltet sind. Am Ende jedes Levels fliegt man in einen kleinen Gang, der ST lädt den nächsten Level und weiter geht es mit purer Action. Das vertikale Scrolling ist zufriedenstellend, das Problem des bisher auf dem ST noch nicht realisierten, horizontalen Scrolling des gesamten Bildschirms hat man durch einen Kompromiß gelöst. Kommt man mit seinem Raumschiff an den Bildschirmrand, gibt es einen kleinen Ruck, so daß der neue Bildausschnitt ins Bild gezogen wird.

Nicht nur die Hintergrundgrafik, sondern auch alle Sprites verfügen über eine schöne Grafik. Sidewinder darf sich neben Goldrunner zu den besten Ballerspielen für den ST zählen.

CBO

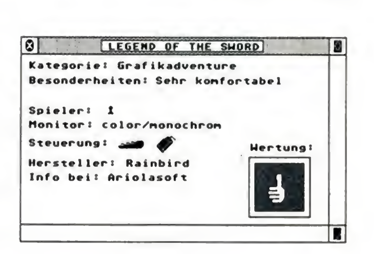


Legend of the Sword



Nach mehr textorientierten Abenteuerspielen wie "The Pawn", "Guild of Thieves" und "Jinxter" bringt Rainbird nun mit "Legend of the Sword" ein Programm mit etwas abgewandeltem Konzept. Anstatt einiger, weniger aber großer Bilder werden nun sehr viele kleine (mehr als 300) Icon-artige Bildchen angeboten, die im Verlauf des Spieles jeweils an passender Stelle eingeblendet werden. Neu sind auch die spielerleichternden Menüs, aus denen alle wichtigen Aktionen über Maus angewählt werden können. Ebenfalls sehr komfortabel wurde das uralte Problem des Kartenzeichnens gelöst: das Programm zeichnet einfach alles mit, was der Spieler gesehen hat. Einige andere sehr angenehme Funktionen sind zum ersten OOPS UNDO- und Ram-SAVE LOAD-Befehle. Zum zweiten kann man über FIND GOTO den Computer Orte bzw. Personen aufsuchen lassen, bei welchen man schon einmal war. Das erspart lange und mühevolle Wege (der Computer sucht die kürzeste Strecke aus), und daß unterwegs während der Suche nichts Wichtiges unbemerkt bleibt, dafür sorgt das Programm. Die Hinweise der Help-Funktion sind zweideutig und hinterhältig wie immer (besondere Warnung vor allzu wörtlichen Interpretationen und vor einfach falschen Informationen!). Einmalig blutig sind die Kampfbeschreibungen und Kämpfe allgemein. Die Geschichte selbst ähnelt wieder einmal der altbekannten Story: Böser Zauberer (Suzar) überfällt Land (Anar), weil magisches Artefakt, welches das Land schützen soll (Schwert & Schild), verschwunden ist. Die Aufgabe liegt auf der Hand: Das Schwert muß her, koste es auch noch so viele tapfere Helden! Zur Unterstützung bekommt man 5 stämmige, kriegserfahrene Begleiter mit, die jedoch meistens nur im Weg rumstehen, sich wertvolle Gegenstände ungefragt aneignen und dumme Bemerkungen fallen lassen. Außer gut prügeln können sie zwar noch einiges mehr, aber um sie dazu zu bringen, gehört schon eine gehörige Portion Glück und Überredungskunst. Das Programm besitzt einen eigenartigen Humor, den man zähneknirschend über sich ergehen lassen muß, und um es zu lösen, benötigt man sicherlich einige Zeit. Es stellt ein neues Konzept dar, ist eine Verbindung zwischen Grafikadventure und Rollenspiel und sollte in keiner Adventure-Sammlung fehlen!

Helmut Scholpp



Leserhitparade August 1988

1. **The Great Giana Sisters (Time Warp)**
2. **Dungeon Master (FTL)**
3. **Star Trek (Firebird)**
4. **Flight Simulator II (Sublogic)**
5. **Bubble Bobble (Firebird)**
6. **Arkanoid (Imagine)**
7. **The Pawn (Rainbird)**
8. **Silent Service (Micro Prose)**
9. **Psion Chess (Psion)**
10. **Tetris (Mirrorsoft)**

Auch im nächsten Monat gibt es wieder eine Leserhitparade. Schreiben Sie uns eine Postkarte oder einen Brief mit Ihren beiden Lieblingsspielen. Wer Lust hat, kann noch dazu schreiben , wie ihm das neue "Relax" gefällt. Unter den Einsendern werden wieder tolle Spiele verlost.

Merlin Computer GmbH
Redaktion ST-Computer
Kennwort Hitparade
Industriestr. 26
6236 Eschborn

ADRESSEN

Ariolasoft
Postfach 1350
4830 Gütersloh 1
Tel: 05241/803871

Leisuresoft
Industriestr. 23
4709 Bergkamen 5
Tel: 02389/6071

Profisoft
Sutthauser Str. 50/52
4500 Osnabrück
Tel: 0541/53905

Rushware Microhandels-gesellschaft
Bruchweg 128-132
4044 Kaarst 2
Tel: 02101/6070

Bomico
Elbinger Straße 3
6000 Frankfurt am Main
90 Tel: 069/706050

Hier die Gewinner der Leserhitparade:

Die fünf "Giana Sisters" gehen an:
Andreas Hoppe, 5840 Schwerte
Alexander Walden, 6246 Schloßhorn
Dirk Ecker, 4440 Rheine
Uys Wunderli, CH-6078 Lungen
Michael Pape, 4780 Lippstadt

Die fünf "Street Gang" bekommen:
Robert Kende, 7000 Stuttgart 1
Rainer Sattler, 7919 Altenstadt
Horst Biletzke, 3000 Hannover 61
Alice Walter, 8501 Feucht
W. Ullenberg, 2841 Wagenfeld

Die fünf "Skyblaster" haben gewonnen:
Ulf Rimkus, 3550 Marburg/Lahn
Manfred Dangel, 8229 Laufe
Bernd Röcker, 8672 Selb
Christian Moehring, 8000 München 2
Stefan Dolezal, 4100 Duisburg



Volker Ritzhaupt
Signum!
Application Systems /// Heidelberg 1988
ISBN 3-9801834-0-8   DM 59,-

Signum!-Zeichensätze
- Ein Buch zur Gestaltung
ISBN 3-9801834-1-6 DM 29,-

Mit den beiden Büchern "Signum!" und "Signum! - Ein Buch zur Gestaltung" bringt die Heidelberger Softwareschmiede Application Systems ihre ersten zwei Bücher auf den Markt. Das geschieht wohl teils wegen den immer wiederkehrenden gleichen Fragen zu Signum!, die der Autor Volker Ritzhaupt (seines Zeichen ein Drittel von Application Systems) unermüdlich zu beantworten versucht, teils aus dem einfachen Aspekt heraus, daß man sich gesagt hat, wenn alle möglichen Buchverlage Bücher zu Signum! herausbringen, warum nicht auch die, die tagtäglich damit zu tun haben und die Probleme der Anwender kennen? Gesagt, getan!

Das erste Buch, Signum!, beginnt ganz von vorne und baut über 427 Seiten bis zur professionellen Nutzung auf. Dazu wurde ein neues Konzept der Computerbuchgestaltung gewählt. Dem Anwender werden zuerst eine ganze Reihe Symbole vorgestellt, die ihn durchs ganze Buch begleiten. Das hat den Sinn, daß man z.B. sofort erkennen kann, ob man Text eingeben oder abspeichern soll, oder ob auf dem Bildschirm etwas ausgegeben wird. Ungewöhnlich, aber sinnvoll!

Anschließend wird kurz auf die Bedienung des ST eingegangen, da gerade Anfänger dabei sehr leicht überfordert sind. Hierbei sei erwähnt, daß man jederzeit Kapitel überspringen kann, wenn man der Meinung ist, daß man das notwendige Wissen schon hat.

Dann beginnt der Einstieg in Signum!. Zuerst werden die grundsätzlichen Vorbereitungen, wie z.B. das Laden von Zeichensätzen, die Einstellung der Parameter und sämtliche Grundfunktionen erklärt. Wer schließlich den Cursor und die Funktionstasten beherrscht, kann sich dem Umbruch von Seiten und Schriftattributen widmen. Schon an dieser Stelle wir man ganz zart in die ersten Geheimnisse der Macroprogrammierung eingeweiht, die in einem späteren Kapitel ausführlich behandelt wird. Die nächsten Kapitel handeln dann von Fußnotenverwaltung, Handhabung mehrerer Zeichensätze und dem Erstellen von Formeln.

Ein weiterer wichtiger Punkt ist das Einbinden von Grafiken. Dazu wird das Screencopy-Accessory erklärt und wie man Grafiken lädt und nachträglich bearbeitet. Danach wird näher auf das Klemmbrett eingegangen, das ja auch bei der Grafik recht brauchbar ist.

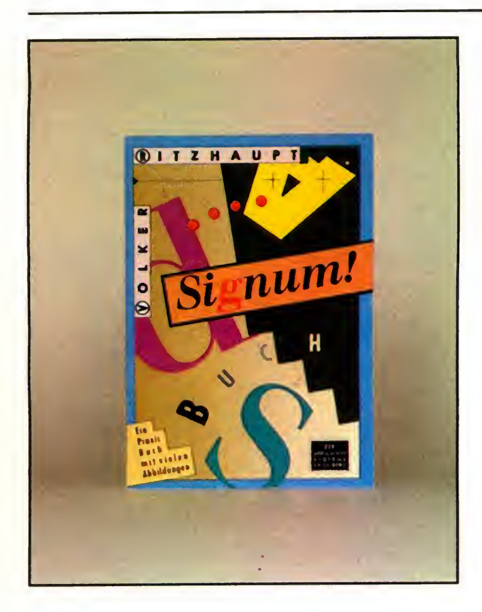
Zuletzt kommt man zu dem Kapitel Spaltensatz, der scherzhafterweise irgendwann mal zu dem Begriff SAP (Signum! Aided Publishing) geführt hat.

Natürlich fehlen im Buch auch nicht die Kapitel Fonteditor und Druckertreiber. Im Anhang kann man dann wichtige Einsteigerhilfen und einen ausführlichen Nachschlageteil finden.

Um ein Fazit zu ziehen: Das Buch ist sowohl für den Anfänger als auch für den Fortgeschrittenen zu empfehlen, denn es beinhaltet jede Menge Tips und Tricks von jemand, der tagtäglich seit knapp 3 Jahren mit Signum! arbeitet.

Zusätzlich ist ein weiteres Buch von Application Systems erhältlich. Es enthält die Signum!-Zeichensätze (inzwischen 351 Stück) und soll zur Gestaltung von Texten dienen. Für einen sehr günstigen Preis bekommt man so auf 224 Seiten eine Übersicht über die erhältlichen Zeichensätze.

HE



Dr. Bernd Enders und Wolfgang Klemme
Das Midi- und Sound-Buch zum Atari ST
Haar b.München, Markt & Technik-Verlag 1988
ISBN 3-89090-528-5
DM 69,-

Der Name sagt wohl schon fast alles über den Inhalt des Buches: Es geht um die Klanggestaltung mit dem Atari-eigenen Soundchip, sowie um Midi, jene digitale Schnittstelle für Musikinstrumente, die, den Entwicklern sei Dank, in den Atari eingebaut wurde.

Nach einer kurzen Einführung in die Geschichte des elektronischen Musizierens geht es gleich in Medias Res: Der Soundchip und seine Programmierung werden behandelt. Dazu gibt es sogar einige Erläuterungen zu akustischen Grundlagen, sehr populärwissenschaftlich zwar, aber für den gegebenen Zweck mehr als ausreichend.

Die Programmierung des Chips wird mit einigen Programmbeispielen in GFA-BASIC und C verdeutlicht. Die Beispiele sind durchweg kurz und einfach geschrieben, leider manchmal etwas zu einfach; man verzichtete zum Beispiel in den C-Listings darauf, die dauernd wiederkehrenden Registernummern als Konstanten zu programmieren, was die Lesbarkeit der Programme doch stark einschränkt. Statt eine Funktion 'Sound' zu definieren, die jeden Ton mit Hilfe einer Parameterliste erzeugt, werden immer wieder listenweis XBIOS-Calls aneinandergehängt, dies allerdings wohldokumentiert. Anfängern hilft diese Programmiertechnik wohl nicht. Die Listings sind auch auf der mitgelieferten Diskette enthalten.

Die nächsten Kapitel erklären die Midi-Schnittstelle. Zuerst wird in einem Kapitel Allgemeines erklärt, dann werden einige Programmbeispiele zur Midi-Programmierung geliefert. In einem späteren Kapitel folgt eine systematische Übersicht über alle definierten Midi-Nachrichten, die auf dem neuesten Stand der Midi-Norm ist. Die Einführungen und Erklärungen zu Midi sind einfach und leicht verständlich gehalten, so daß auch Neulinge in der Materie sich ohne weiteres zurechtfinden sollten.

Mit Hilfe der systematischen Erklärungen sollte man auch in der Lage sein, Verständnisschwierigkeiten in einem Midi-Setup zu beheben und überhaupt etwas besser zu verstehen, was in den Midi-Kabeln zwischen den einzelnen Instrumenten so kreucht und fleucht.

Die Programmier-Beispiele sind nützlich, erwarten Sie aber nicht, mit dem in diesem Buch enthaltenen Know-How einen kompletten, professionellen Sequencer schreiben zu können. Zwar ist genug enthalten, um über die BIOS-Aufrufe an die Midi-Schnittstelle zu kommen, doch gehören zu einem Sequencer-Programm z.B. auch komplexe interne Datenstrukturen und einiges Wissen über interruptgesteuerte Programmierung, deren Erläuterung nicht Sache dieses Buches ist.

Ein weiteres Kapitel ist professionellen auf dem Markt befindlichen Midi-Produkten, die auf Basis des Atari arbeiten, gewidmet. Einige der wichtigsten Programme werden vorgestellt, auch ein 16Bit-Stereo-Sampler für den ST. Solche Beschreibungen haben naturgemäß meist den Nachteil, bereits bei Erscheinen des Buches nicht mehr die gerade aktuellen Versionen der beschriebenen Programme zu behandeln, so auch hier.

Hardware-Bastler kommen auch auf ihre Kosten: Im Anhang wird der Selbstbau eines Audio-Digitizers beschrieben, der es ermöglicht, Klänge mit einer Auflösung von 8 Bit in den Speicher des Rechners zu befördern. Die dazugehörige Software, die natürlich auch auf der Programmdiskette enthalten ist, erlaubt die Wiedergabe dieser Klänge über den Monitor-Lautsprecher des ST, allerdings, der Beschränkungen des Soundchips des ST wegen, nur mit 6 Bit Auflösung.

Im Anhang sind noch einige etwas kompliziertere Midi-Programme zu finden, ein primitiver Sequencer etwa oder eine Grafik-Demo, die auf einfache Weise Noten auf dem Bildschirm zeigen kann. Auch ein Demo-Editor-Programm für den Korg DW8000-Synthesizer ist dabei. Die Besipiele des Anhangs sind zum größten Teil in GfA-Basic und C dabei, manches aber auch nur in einer Sprache.

Abgerundet wird der Anhang durch einige Tabellen, z.B. für die Frequenzen des Soundchips, für Hex-/Dez-Umwandlung, für die Umrechnung von Soundchange-Nummern in die Preset-Nummern verschiedener Hersteller usw.

Das Buch ist insgesamt leicht lesbar geschrieben und ertklärt gut, wenn auch nicht immer ganz exakt. Die Illustrationen sind ganz hervorragend und mit viel Mühe gezeichnet. Die Programmbeispiele erfüllen ihren Zweck, könnten jedoch (programmier-) pädagogisch besser sein. Alles in allem: ein empfehlenswertes Buch.

CS



PASCAL-Programmierung auf dem ST

Ich besitze einen Atari Mega ST2 und den PASCAL-Compiler von CCD. Die Beschreibung der GEM-Funktionen im Handbuch des Compilers ist zwar sehr umfassend, setzt aber leider GEM-Kenntnisse voraus, die ich als 'GEM-Anfänger' nicht besitze.

Meine Frage an Sie ist, ob Ihnen Bücher bekannt sind, die die GEM-Routinen für Anfänger verständlich, und zugleich die Handhabung vom CCD-Pascal aus beschreiben. Während meiner bisherigen Suche habe ich zwar mehrere Bücher über GEM entdeckt, die sich aber alle auf C als Programmiersprache beziehen.

(Stefan LambertSpiesen-Elversberg)

Red.: Leider ist der Markt an Büchern, die sich mit PASCAL auf dem ST beschäftigen, relativ klein. Zunächst möchten wir Sie darauf hinweisen, daß CCD seine Dokumentation zu PASCAL 2.0 völlig überarbeitet hat und sie ist zu einem Preis von 49,- DM plus 5,- DM Versandkosten für eingetragene Pascal-Plus-Benutzer bei CCD zu beziehen. Das einzige Buch, das wir ausfindig machen konnten und das sich mit PASCAL-Plus beschäftigt, geht nicht besonders stark auf GEM ein: es ist von Peter Wollschläger und heißt ATARI-ST-Programmierpraxis ST-Pascal. Da ST-PASCAL in der 2.0-Version allerdings alle GEM-Routinen vordefiniert hat, und diese im Aufruf praktisch wie in C aussehen, würden wir vorschlagen, sich eines der sehr guten C-angelehnten Bücher wie zum Beispiel "Softwareentwicklung auf dem ATARI ST" aus dem Hüthig Verlag (Buchbesprechung Mai 1987) oder "Das Profibuch" aus dem Sybex-Verlag (Buchsprechung März 1988) anzusehen.

*

Accessories sind aber in Assembler möglich?!

Ich lese in der ST-Computer Februar '88, daß es unmöglich sei, ein Accessory in GFA-Basic zu erstellen. Schon gut, aber es ist mit Assembler zweifellos möglich. Würden Sie bitte so gut sein und ein Accessory in Assembler abdrucken? Das Programm müßte ja nur "Hello" auf den Bildschirm schreiben. Was wichtig ist, ist, wie man ein solches Accessory initialisieren muß. Nur noch ein Wort 'Congratulations for the quality of your magazine. As you can see, you have readers far from home'.

(Bernard Boust, Le Petit-Quevilly)

Red.: Merci beaucoup! We appreciate having readers far from home and like to thank all those who send us letters from foreign countries, even Bolivia is one of them. Back to german: Sie haben recht, wir werden demnächst in einer Folge der ST-Ecke für einige Programmsprachen ein kleines Accessory-Skelett darlegen. Da an dieser Stelle Assembler zu umfangreich ist, hilft Ihnen vielleicht aber ein C-Skelett weiter (s.Listing).

*

IMAGIC-Nachlese

Nachdem ich die neueste Ausgabe Ihrer Zeitschrift gelesen habe, habe ich einige Bemerkungen, die ich unbedingt loswerden muß. Daher schreibe ich Ihnen diesen Leserbrief:

Mich irritiert in zunehmendem Maße die Inkompatibilität verschiedener Programmier- und Entwicklungssysteme. Ich beziehe mich hier auf die unterschiedlichen Objekt-File-Formate...Auf dem IBM setzt z.B. Microsoft Maßstäbe, auf dem Atari ST dagegen...? In Ihrem Artikel über "Imagic" schreiben Sie, daß es zwar möglich, aber nicht sinnvoll ist, mehr als 71 Bilder pro Sekunde im Rechner aufzubauen. Ich muß Sie korrigieren: Es ist allein schon von der Rechengeschwindigkeit des 68000 her unmöglich, ganze Bilder in dieser Zeit aufzubauen. Eine kleine Rechnung:

Der Bildschirmspeicher des ST hat immer 32000 Bytes. Der einfachste und schnellste Befehl zur Beeinflussung dieses Speichers ist im Prinzip MOVE.L Dn,(An)+; er dauert 12 Taktzyklen. Dazu kommt dann noch ein Branch, der mindestens 10 Taktzyklen benötigt. Diese kleine Schleife muß bei 8 MHz 8000 mal durchlaufen werden und dauert daher 8000* (12+10), also 176000 Taktzyklen, gleich 0.022 Sekunden. Das sind etwa 45 Bilder pro Sekunde. Es ist also nur möglich, Teile eines Bildes in weniger als 1/71 Sekunde aufzubauen, aber niemals ein ganzes Bild.

Angesichts der obigen Rechnung erscheint es mir auch zweifelhaft, ob sich wirklich ganze Bilder in 1/25 Sekunde dekomprimieren lassen. Ungepackt lassen sich in einem ST mit 1 MB auch höchstens 32 Bilder lagern, wenn man wirklich alles über Bord wirft. Das gibt etwa eine halbe Sekunde Bildervergnügen. Daher ist Ihre Beschreibung von "IMAGIC" zumindest an dieser Stelle sachlich unkorrekt. Es ist wahrscheinlicher, daß IMAGIC Bilderfolgen im Speicher hat und nur die Unterschiede von Bild zu Bild speichert und entsprechend packt. Hätte man 1000 komplett unterschiedliche Bilder im Speicher, käme etwa alle Sekunde ein Bild auf den Bildschirm, da so dichtes Packen viel Zeit braucht - hier gilt eine alte Regel: Der Speicherverbrauch ist umgekehrt proportional zum Zeitverbrauch. Nur mit einem Blitter ginge der Bildschirmaufbau schneller. Leider haben Sie in ihrem Test vergessen, den Punkt zu erwähnen, ob der Blitter unterstützt wird oder, wie in so vielen Programmen, nicht.

Red.: Zunächst einmal möchten wir uns bedanken, daß Sie sich so viel Mühe in der Be-

```
extern gl_apid;
main()
{
  int msgbuff[8]; /* Ereignis-Puffer */
  appl_init();    /*Initialisierung des GEM-AES */
 menu_id=menu_register (gl_apid,"Mein Eintrag");
  while(1) /* Accessories enden nie ! */
   {
    evnt_mesag(msgbuff); /* Ist unsere Accessory gemeint ? */
    if (msgbuff[0]==AC_OPEN && msgbuff[3]==menu_ud)
       { /* hier kommt ihr Programm hin.*/
         printf("Hello\n");
       }
   }
}
```

C-Skelett für die Programmierung eines Accessorys

rechnung der Zeiten gemacht haben. Leider müssen wir Ihnen aber mitteilen, daß Sie nicht ganz recht haben. Der Artikel von IMAGIC war gut recherchiert, allerdings haben wir uns aufgrund Ihres Leserbriefes noch einmal mit den Programmierern unterhalten. Um es vorwegzunehmen: Alle Aussagen, die im Artikel gemacht wurden, sind korrekt.

Zunächst möchten wir zur Bildkopierroutine Stellung nehmen: Die einfachste Routine wäre theoretisch mit dem Befehl MOVE.L (An)+,(An)+ zu bekommen und nicht, wie Sie schreiben, mit MOVE.L Dn,(An)+, da dieser Befehl den Bildschirm nur mit einem Wert füllen würde. Daher wird die Routine noch langsamer, was für Sie sprechen würde, weil derBefehl MOVE.L (An)(An)+ 20 Zyklen + 10 Zyklen eines DBRanches sogar 2400000 Zyklen ergeben würde. Trotzdem ist es über folgenden Trick möglich, die Routine enorm zu beschleunigen: Benutzt man den Befehl MOVEM.L (An)+ ,Registerliste, so läßt sich eine Routine schreiben, die nur etwa 148000 Zyklen (entspricht etwa 54, abzüglich der Interrupts 50 Bilder) benötigt und etwa folgendermaßen aussieht:

```
movem.l (A0)+, Registerliste
movem.l Registerliste, (A1)
lea     distanz (A1),(A1)
dbra    reg, label
```

Allerdings werden Sie feststellen, daß im Artikel nicht steht, daß 71 Bilder pro Sekunde aufgebaut, sondern dargestellt werden, was ein großer Unterschied ist. Wie Sie als Programmierer wissen, kann man beim ST durch einfaches Ändern eines Zeigers einen anderen Bildschirminhalt einblenden, was theoretisch schneller (aber wie erwähnt nicht sinnvoll) geht als 71 mal in der Sekunde! Ungepackte Sequenzen sind daher mit einer so hohen Geschwindigkeit möglich, wobei dies bei einem 4 MB-Rechner zu einer Sequenz von etwa 2 Sekunden führen würde. Möchte man längere Bilderfolgen, muß man sich damit begnügen, mit gepackten Bildern zu arbeiten. Die schnellste und speichergünstigste Art dabei ist, wie Sie schon richtig bemerkten, die Differenzkomprimierung, die sich besonders bei nicht allzu starken Unterschieden zwischen verschiedenen Bildern auszahlt. Nur bei dieser Methode ist auch möglich, über tausend Bilder in einen MEGA ST4 zu bekommen - trotzdem beachtlich, denn das bedeutet für jedes Bild 4 Kilobyte. Die Dekomprimierung dieser differenzkomprimierten Bilder ist, da hier nicht ganze Bilder bewegt werden, tatsächlich mit bis zu 25 Bildern (je nach Komplexität) möglich, was mir die Firma IMAGIC Grafik ausdrücklich bestätigte - machen Sie die Probe aufs Exempel. Eine sachliche Unkorrektheit liegt also nicht vor! Betrachtet man das Beispiel von völlig unterschiedlich gepackten Bildern, so wird bei Imagic immer noch eine Geschwindigkeit von 5-6 Bildern pro Sekunde erreicht (minimal 4 Bilder!) und nicht, wie Sie schätzten, von einem Bild. Die Zeiten sind übrigens alle ohne Blitter. Der Blitter wird von Imagic unterstützt, wobei es übrigens so ist, daß Bildausschnitte, die nicht mehr als ein Viertel des Bildschirms ausmachen, schneller über eigene Routinen bewegt werden, als wenn man die BitBlt-Routine des Betriebssystems nehmen würde, die den Blitter benutzt. Imagic repariert als erstes Programm sogar die ab und zu vorkommenden Busfehler des Blitters (siehe ST-Ecke, ST-Computer Juni '88).

Wir hoffen, Sie davon überzeugt zu haben, daß alles seine Richtigkeit hat, und nicht aus der hohlen Hand gegriffen ist. Sollten Sie dennoch Zweifel haben, so wird sie Ihnen die Firma IMAGIC Grafik in einem Gespräch alle nehmen, außerdem läßt sich alles ausprobieren.

Zu dem Problem der Inkompatibilitäten der Entwicklungssysteme: Selbst auf dem MS-DOS-Sektor gibt es unterschiedliche Formate (nehmen Sie nur TURBO-C und Microsoft-C), so daß es dort auch nicht unbedingt besser aussieht als auf dem ATARI ST. Auf dem ST gibt es aber einen Link-Standard, der durch das Entwicklungssystem von Digital Research, das von ATARI vertrieben wird, gesetzt wurde. An dieses Format hat sich zum Beispiel CCD-Pascal gehalten, und MEGAMAX 2.0, das demnächst auf den Markt kommt, wird dieses Format auch haben - wie noch ein paar andere Systeme. Wir sind dabei, die Formate zusammenzutragen (das alte Megamax-Format ist übrigens in den Unterlagen dargestellt), aber diese Serie wird wahrscheinlich noch ein wenig auf sich warten lassen. Umwandlungsprogramme sind wahrscheinlich schwierig, da verschiedene Formate unterschiedliche Stackbehandlung von globalen und lokalen Variablen haben. Sollten Sie Entwickler sein, so wird Ihnen ATARI sicherlich bereitwillig Information über dieses Format zukommen lassen. Nebenbei: wir haben vor einigen Monaten sogar einen Linker veröffentlicht.

*

Dialogverarbeitung und Accessories in GFA-BASIC

Ich möchte bezüglich der Dialogboxenverwaltung in GFA-BASIC einige Fragen an Sie richten. Kann man die "FORM-DO"-Routine umgehen? Auch wenn in der Dialogbox nichts passiert, kann man keine anderen Programmschritte mehr ausführen lassen. Mein Ziel: Der Rechner soll Daten verarbeiten und gleichzeitig Ereignisse in der Box beachten (Interrupt?).

Können in GFA-BASIC auch Accessories geschrieben werden und wenn ja, wie?

Ist es möglich, eine Hardcopy von GFA-BASIC so zu beeinflussen, daß der Drucker eine Zeile doppelt ausdruckt, um ein dunkleres Bild zu erreichen?

Können Sie mir eine Aufschlüsselung aller GEMSYS schicken?

(Rudolf Hausmann, Steinbach)

Red.: Wie in der ST-Computer-Ausgabe 7/87 beschrieben worden ist, kann man sich auch eine eigene Form_Do-Routine 'zurechtstricken'. Das dabei gezeigte Beispiel ist zwar in C geschrieben, es spricht aber nichts dagegen, diese Routine auch in GFA-BASIC zu realisieren. Verzichtet man auf Edit-Felder, so läßt sich die Bearbeitung stark vereinfachen, da man dann nur noch mit objc_find() aus den gegebenen x/y-Koordinaten das Objekt ermitteln muß. Eine Interruptbearbeitung ist schwierig, da AES- und VDI-Routinen des GEM nicht reentrant sind und daher aus dem Interrupt nicht aufgerufen werden sollen.

Accessories sind bisher mit GFA-BASIC noch nicht möglich. Wie von GFA aber zu erfahren war, wird darüber nachgedacht, in den neuen zu GFA-Basic 3.0 gehörenden Compiler entsprechende Optionen einzubauen. Da dies aber nicht sicher ist, und man sicher noch ein knappes halbes Jahr auf den Compiler warten muß, heißt die Antwort "NEIN".

Will man eine Hardcopy-Routine mit doppelter Zeile ausdrucken, so kann man sich sicherlich eine eigene Routine schreiben. Eine einfache Manipulation ist, soweit wir wissen, nicht möglich. Vielleicht weiß ein Leser mehr?

Verzeihen Sie, aber Ihnen eine Aufschlüsselung der GEM-Befehle Ihnen zuzuschicken, ist leider nicht möglich, bedenkt man, wieviele Befehle GEM enthält. Selbst eine reine

Angabe der Befehle mit Parameter würde viele Seiten füllen. Außerdem sind die Befehle (auch mit Parametern) ohne ausführliche Erklärung bei GEM nicht sinnvoll. Es gibt aber genügend umfangreiche Bücher, die sich mit diesem Thema beschäftigen und angemessene Preise haben.

*

MEGA-Tastatur

Kann man an den ST520+ eine neue MEGA-Tastatur anschließen? Mein Rechner wurde in ein PC-Gehäuse eingebaut und die neue Tastatur soll doch sehr gut

Thomas Linkmacher

Red.: Zugegeben, die neue MEGA-ST-Tastatur ist um einiges besser als die der kleineren Brüder, auch der Anschluß ist theoretisch möglich, die Tastatur wird aber normalerweise nicht einzeln verkauft. Allerdings könnten Sie probieren, sie als Ersatzteil bei Ihrem ATARI-Händler zu beziehen. Als Alternative gibt es aber inzwischen Interfaces (Schnittstellen), die es ermöglichen, PC-Tastaturen an den ST anzuschließen, und das sogar mit relativ kleinem Aufwand.

*

Anfänger

Ich bin Anfänger, und für mich stellen sich folgende Fragen: Wie werden Programmlistings in höheren Programmiersprachen in den ST eingegeben? Braucht man für jede Programmiersprache einen eigenen Compiler? Mit welcher Sprache soll man als Einsteiger anfangen, um richtiges Programmieren zu erlernen? Gibt es zu diesen Sprachen Übungsbücher mit Lösungen, um sich zu kontrollieren?

(Josef David, Wenden 1)

Im Unterschied zu Home-Computern, wie zum Beispiel dem C-64 oder dem ATARI XL, ist die Eingabemöglichkeit einer Programmiersprache in einem Personal Computer wie dem ATARI ST nicht integriert, sondern muß als Programm geladen werden. Ein solches Programm nennt sich Editor und ist eine Art Textverarbeitung, die es ermöglicht, Programmtexte möglichst schnell einzugeben. Furore wegen seiner Schnelligkeit hat in diesem Zusammenhang etwa vor einem Jahr der Editor TEMPUS gemacht.

Leider braucht man für jede Programmiersprache einen eigenen Compiler, da dieser auf die jeweilige komplexe Art derselben abgestimmt werden muß - das liegt nun einmal in der Natur der Sache.

Mit welcher Sprache man anfangen soll, darüber streiten sich bekanntlich die Gelehrten. Früher hieß es BASIC (Beginners All Purpose Symbolic Instruction Code - Für Anfänger allgemein geeignete Sprache), allerdings verführt dieses allzusehr zum unstrukturierten Programmieren. Daraufhin wurde von Niklaus Wirth PASCAL für seine Studenten entwickelt, was auch bald auf viele Rechnern implementiert wurde.

Eine weitere sehr schöne aber zugegeben für den Anfänger etwas schwierige Sprache ist C, die "Muttersprache" des ATARI ST. Die neueste Welle ist MODULA 2, was eine Weiterentwicklung in Richtung auf noch stärkere Modularität und Strukturierung in Bezug auf PASCAL darstellt, aber selbst BASIC erlebt durch seinen Verlust der Zeilennummern und der Vermeidung der so verpönten GOTOs eine Wiedergeburt und hat heute schon starke Ähnlichkeit mit PASCAL. Auf Universitäten und Fachhochschulen wird heute hauptsächlich PASCAL und C gelehrt, PASCAL setzt sich auch in den Schulen durch.

Wenn ich heute noch einmal beginnen müßte, würde ich es wahrscheinlich mit MODULA 2 versuchen oder, falls man keinen Lehrer zu Verfügung hat, mit BASIC, da es darüber die meisten Bücher gibt. Achten Sie aber darauf, daß Sie schon von früh an auch in BASIC strukturiert programmieren und vielleicht gleich mit einem BASIC arbeiten, welches überhaupt keine Zeilennummern, sondern nur noch Prozeduren besitzt.

Gerade im BASIC-Bereich gibt es sehr viele Bücher, die auch teilweise für das Selbststudium mit Fragen und Antworten gedacht sind. Leider sind diese noch oft auf das alte BASIC ausgelegt, also besser einmal schauen, ob es schon etwas Neueres in dieser Richtung gibt.

*

Wie funktioniert das RCS?

Es würde mich sehr freuen, wenn Sie einmal einen Artikel über die RCS-Diskette bringen könnten. Denn darauf ist zwar eine Beschreibung über die Programmänderungen, nicht aber, wie man überhaupt ein Programm eingeben kann, was doch das Wichtigste ist. Was nützt mir sonst die Diskette? Vielleicht wäre so ein praktischer Artikel für manchen interessant.

(Wilhelm Leurs, Bonn)

Red.: Richtig! Das haben wir uns auch gedacht, als wir unser zweites Sonderheft herausbrachten. Deshalb haben wir den ersten Artikel des Sonderheftes einer Einführung in das RCS und den zweiten der Verwendung der vom RCS erzeugten Dateien gewidmet. Ich bitte Sie daher, einmal in unser zweites Sonderheft hineinzuschauen; dort werden Sie alle wichtigen Informationen bezüglich der Handhabung des Resource Construction Sets finden. Übrigens: Im RCS können Sie keine Programme eingeben, sondern nur die bekannten Dialogboxen und Menüleisten erstellen. Die dann abgespeicherte Datei kann später in eigenen Programmen verwendet werden.

Probleme mit dem MEGA ST

Ich hatte ein Jahr lang einen 1040 ST und jetzt, seit einem halben Jahr den MEGA ST4. Es gibt sehr interessante Programme, die aber leider auf dem MEGA ST nicht funktionieren. Wo liegt da der Fehler, und kann man etwas machen, daß auch diese Software wieder anzuwenden ist? Vielleicht können Sie mir weiterhelfen.

Red.: Wahrscheinlich haben sich die Programmierer bei ihrer Arbeit leider nicht an die von ATARI strikt gegebenen Regeln gehalten und haben zum Beispiel undokumentierte oder für spätere Betriebssystemversionen nicht festgelegte Adressen benutzt. Die beste Methode ist, sich bei den entsprechenden Firmen zu beschweren und auf Abhilfe zu pochen. Eine gute Firma kennt das Problem sowieso, hat schon längst für Abhilfe gesorgt und wird Ihnen sofort behilflich sein. Solange kann ich Ihnen nur folgenden Tip geben: Es ist auch bei eingebauten ROMs möglich, das Betriebssystem von der Diskette zu booten, so daß es sich anbietet, das alte Betriebssystem von der Diskette zu laden und nicht das der ROMs zu benutzen. Da Sie einen MEGA ST4 besitzen, spielt Speicherplatz bei Ihnen nun wirklich keine Rolle. Natürlich löst dies das Problem nicht bei Programmen, die gebootet werden müssen; da hilft nur der rettende Griff zum Briefpapier oder zum Hörer.

Wenn auch Sie Probleme haben oder Anregungen zu den Leserbriefen geben wollen, schreiben Sie an folgende Adresse:

Merlin Computer GmbH
"Leserbrief ST"
Industriestr. 26
6236 Eschborn

Wir versuchen Ihnen dann zu helfen.

PR8-SOFT

Telefon: 0931/464414

DAS BACKUP-PROGRAMM FÜR DEN ATARI ST

PROCOPY ST

- *Fertigt Sicherheitskopien von geschützten Disketten auf Tastendruck, ohne aufwendige Menüs*
- *Selbst der härteste Kopierschutz ist kein Problem*
- *Kopieren in einem Durchgang, Mehrfachkopien ohne Neueinlesen des Originals*
- *Verify-Option*
- *Analyse-Option*
- *Schnellformatier-Routine*
- *Ständiger Updateservice für neue Kopierschutzarten*

hi 88

Schützen Sie Ihre Software, arbeiten Sie nur mit Backups, PROCOPY machts möglich.
Die Low-Cost Lösung für jeden ATARI ST-User.

Mit ausführlichem Handbuch **nur DM 99.-**

Telefonische Eil-Bestellung (24 Stunden-Service): 0931/46 44 14
PR8-SOFT Klaus-M. Pracht · Postfach 500 · D-8702 Margetshöchheim

Lieferung per Nachnahme (Versandkosten DM 5.- + NN-Gebühr) oder gegen Scheck (+Versandk. DM 5.-). Auslandslieferungen gegen Scheck (+Versandkosten DM 10.-)

STARKE SOFTWARE FÜR STARKE COMPUTER

TKC-EINNAHME/ÜBERSCHUSS PLUS *(Buchführung)* **DM 149,--**
Automatische Führung von MWSt.-Konten. Saldenlisten, Kontenblätter, Bilanz, USt.-Voranmeldung. Bis zu 5 MWSt.-Sätze. Abschluß wahlweise Monat, Quartal oder Jahr. Universelle Druckeranpassung, incl. ausführlichem Handbuch !

TKC-HAUSHALT PLUS *(Haushaltsbuchführung)* **DM 129,--**
Dauerbuchungen, Bilanz, Kontenblätter, Saldenlisten mit Teilsummen, Monats- und Jahresabschlüsse, frei erstellbarer Kontenrahmen (max. 250 Konten), universelle Druckeranpassung, ausführliches deutsches Handbuch !

TKC-TRAINER *(Trainingsprogramm für Alles und Alle)* **DM 99,--**
Trainingsprogramm der Superlative! Geeignet für Deutsch, Mathematik, Vokabeln und anderen Lernstoff. 5000 Datensätze pro Datei, Berücksichtigung von Mehrfachbedeutungen bei Vokabeln, Zufallsgenerator, Auswertung. Lernen auch Sie nach dem **KARTEIKASTEN-PRINZIP**. Incl. ausführlichem Handbuch !

TKC-BANKMANAGER *(Verwaltung von Bankformularen)* **DM 99,--**
Getrennter Aufbau von Bank- und Adressdatei. Mischen von Banken und Adressen über Auftragsmaske. Geeignet für Überweisungen, Schecks, Zahlkarten, etc. Freie Anpassung an jedes Formular mit einfachem Texteditor. Anpassung an jeden Drucker. Sammler für getätigte Überweisungen. Incl ausführlichem Handbuch !

TKC-VIDEO *(Verwaltung von Videofilmen)* **DM 79,--**
Verwaltet bis zu 5000 Videofilme pro Datei. Umfangreiche Sortier- und Suchfunktionen. Ausdruck von Listen und Etiketten. Incl. Handbuch !

TKC-MUSICBOX *(Verwaltung von MC's, CD's und LP's)* **DM 79,--**
Verwaltet bis zu 5000 LPs, CDs oder MCs pro Datei. Titel-Suchfunktion, Ausdruck von Listen und Etiketten, Sortierung nach LP-Titeln, Druckeranpassung !

ST-MATHETRAINER *(Lernprogramm für 1.-6. Schuljahr)* **DM 59,--**
1x1, Grundrechnen, Umrechnung von Gewichten und Längenmaßen, Benotung

ST-RECHTSCHREIBEN *(Lernprogramm für 1.-6. Schuljahr)* **DM 59,--**
Singular & Plural, Interpunktion, Schwerpunktdatei für falsche Antworten !

ST-VOKABELTRAINER *(Trainingsprogramm für Vokabeln)* **DM 49,--**
Für die Sprachen Englisch, Französisch, Italienisch und Spanisch! Sonderdatei!

ST-GIRO PLUS *(Druckprogramm für Zahlungsträger)* **DM 49,--**
Bedruckt Überweisungsträger und Lastschriften, Anpassung über einfachen Editor!



TK COMPUTER-TECHNIK
BISCHOFSHEIMER STRASSE 17 • 6097 TREBUR-ASTHEIM
TELEFON : 06147 / 550

AB COMPUTER

AMIGA · ATARI
PC kompatibel

A. BÜDENBENDER
Wildenburgstraße 21
5000 Köln 41
Telefon (02 21) 4 30 14 42

IHR FACHHÄNDLER IN KÖLN FÜR AMIGA / ATARI / PC
Wir bieten Ihnen noch Beratung und Service für Ihren Computer

AB Doppelfloppy 2★726KB graues Metallgehäuse o. Schrauben an den Seiten, externe Stromversorgung Spitzenqualität mit NEC FD1037a eigene Herstellung ST 314 kompat. **548,–**
AB Einzelfloppy 1★726KB mit NEC FD 1037 noch kleiner 25 mmh ★ 170 mm T. mit externem Steckernetzteil komplett anschlußfertig die kleinsten z. Zeit **279,–**
AB mit FD1037 Einzellaufwerk mit Buchse zum Anschluß eines 2. Lw. 3.5 Zoll oder 5.25 Zoll **319,–**
AB 5.25 Zoll Laufwerk mit 80/40 Trackumschaltung Pc Ditto kompatibel **398,–**
AB 5.25 Zoll + 3.5 Zoll Lw. in einem Gehäuse int. Netzteil AB/BA umschaltbar **648,–**
NEC FD 1037 NEU 3.5 Zoll 5V Vers. 28★140 mmH ★ T ohne Gehäuse roh Lw. **198,–**
ST Floppystecker ... 6,– Buchse .. 8,– Monitor Stecker St. ... 6,– Monitor Buchse .. **8,–**
ST Floppykabel fertig für Lw. A .... 25,– Lw. A+B .... 30,– Steckernetzteil Floppy 5V **35,–**

EIZO Multimonitor beste Qualität für St alle drei Aufl. 0.28 Dot. **SUPER** **1498,–**
kein Durchlaufen mehr beim Umschalten Monitor 820X620 Auflösung
Farbmonitor für St mit Scart .... 598,– HF Modulator für ST steckbar mit Netzteil **198,–**
Monitor Kabel für Multisync ...... 75,– Scart Kabel fertig 1,5m ...... 39,– Scart 3 m **49,–**
Switchplatine2 2 Monitore an 1 ST ...... 35,– Uhr für ST mit Blitter Tos keine Software **105,–**
SM 124 Monitor schwarzweiß .... 430,– Multisync GS NEC alle 3 Auflösungen an St **598,–**

ST Speichererweiterung 512 KB für 260/520 STM Computer steckbar 350,–

ST Epromsatz 27256 pro Stk. ..... 12,– Rom Satz St ..... 98,– U7 2★schneller Laden **25,–**
Eprommer für Romport komplett für alle Eproms mit Software anschlussfertig **149,–**
Easyprommer von Merlin für Druckerport komplett mit Software für Druckerport 189,–

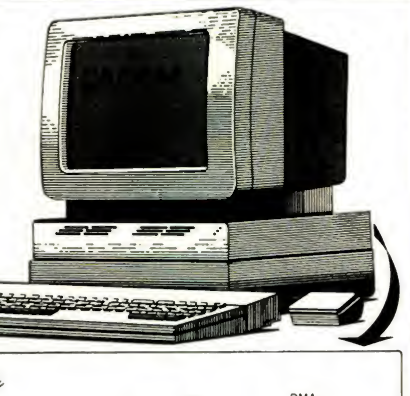
Festplatte 40 MB 28ms SH 205 mit Seagate 251 mit Turbo DOS einer der schnellsten Platten **1798,–**
SH 205 20 MB Festplatte .... 998,– **Vortex HD 20 MB .... 1100,–** Vortex 30 MB **1298,–**
Vortex HD 60 MB komplett anschlußfertig .... **2100,–** Vortex HD 100 MB komplett 3500,–
alle Vortex Platten mit Backup Prg. Cache / Park Prg. sowie Turbo DOS läuft jetzt!
ST 1040 + SM 124 + Maus .... auf Anfrage Mega St 2 Mega St 4 .... auf Anfrage
Scanner 400★400 Dpi komplett **mit Software für Laser** anschlußfertig **2998,–**
Atari PC XT 3 auf Anfrage / Ega VGA Karten / Festplatten PC / At XT .... auf Anfrage

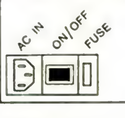
NEC P6 Plus NEU 85 KB Buffer 255 Zeichen sek. mit eingeb. Traktor Spitze **1850,–**
NEC P2200 NEU 24 Nadeln voll P6 kompatibel inkl. Traktor/Einzelblatt **948,–**
STAR LC 10/24 Nadeln 154 Sek./Z. 360★360 voll P6 kompatibel mit Traktor **1100,–**
STAR LC 10 Drucker 9 Nadeln deutsches Handb. Color .... 748,– Mono schw. **619,–**
ATARI Laserdrucker 8 Seiten pro Minute komplett anschlußfertig .... auf Anfrage
Telefax Schneider komplett anschlußfertig .... 2598,– NEC NEFAX 10 .... **2998,–**

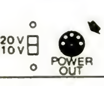
Adimens 2.1 Datenbank ... 195,– Signum 2 ... 378,– Stad Grafik ... 159,– PC Ditto ... **179,–**
ST Pascal 2.00 plus CCD .. 249,– 1 St Adress .. 139,– BTX ST .. 298,– 1 St Word .. **180,–**
Freesoftware alle Prg. aus St Computer pro Stk. .... 8,– Versand innerhalb 48 Stunden
Disk 2DD Fuji ..... 35,– Fuji 1DD ..... 28,– 2dd No Name ..... 24,– Software Liste anfordern

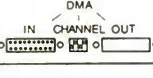
Wir liefern für Ihre Firma die richtige Soft/Hardware / Beratung nach Wunsch
Händleranfragen erwünscht. Die Preise sind unverbindl. empf. Verkaufspreise.
Atari / Star NEC Schneider sind eingetragene Warenzeichen – Versand ins Ausland nur per Vorkasse / Überweisung aufs Konto
Telefax 02 21 / 4 30 16 95 · Bürozeiten 10 – 13 Uhr 14 – 18 Uhr · Samst. 10 – 14 Uhr.

21MB 1198,- DM
32MB 1348,- DM
42MB 1798,- DM
65MB 1998,- DM
85MB 2498,- DM
98MB 2798,- DM
128MB 3498,- DM









Die LACOM HD's bieten nicht nur den Standard an Komfort, sondern noch einiges mehr. So z.B. die herausgeführte DC-Stromversorgung, an der weitere Geräte (Rechner, Floppy's, Erweiterungen) angeschlossen werden können. Die HD und alle angeschlossenen Geräte werden über nur einen Schalter geschaltet !

Die LACOM HD's lassen sich problemlos mit einem zusätzlichen Laufwerk aufrüsten. Hard- & Software sind bereits dafür ausgelegt !

Die Bootpartition kann beim Systemstart aus einer der acht gewählt werden. Somit stehen acht verschiedene Bootkonstellationen für die verschiedensten Anwendungen zur Verfügung.

Eine Zugriffszeit von 40ms und ein echtes Head-Autopark (parkt selbst bei Stromausfall die Köpfe) sind weitere Leistungsmerkmale der LACOM HD's !

Der Lieferumfang: betriebsfertige HD, Software (u.a. HD-Utility von A.S.H.), ausführliches deutsches Handbuch, 1 m abgeschirmtes Buskabel, 2 m Netzkabel !

Die Komponenten sind einzeln erhältlich:
- Gehäuse (bearbeitet) 149,-
- " aufgebaut (Lüfter, Netzteil) .. 298,-
- PC-Netzteil 150W 149,-
- Adapter mit Software ... ab 149,-
- OMTI's - Laufwerke - Stecker ! ☎

! 44MB Wechselplatten !

LACOM // Computer & Musik
Ilias Lazaridis - Emscherstr. 45
4200 Oberhausen 11 ☎ **0208 / 65 06 69**





INSERENTEN-VERZEICHNIS

LASER C

Megamax, die Zweite

Das neue C-Entwicklungssystem von Megamax heißt Laser C. Wir gehören zu den Glücklichen, denen eine englische Testversion mit englischem Handbuch vorliegt. Wie das bei solchen Testversionen üblich ist, traf sie erst kurz vor Redaktionsschluß ein und Application Systems /// Heidelberg, der deutsche Distributor versicherte uns, daß die endgültige Version ganz anders und natürlich viel besser, deutscher und fehlerfreier sei.

Was ist neu an Laser C gegenüber dem bekannten Megamax C? Alles, oder besser gesagt, fast alles. Das Resource Construction Programm (RCP.PRG) trägt jetzt die Versionsnummer 2.1, ist aber kaum verändert worden. Auch an der Tatsache, daß ein Ordner mit Namen MEGAMAX in der obersten Ebene des Dateiverzeichnisses stehen muß, hat sich nichts geändert. Alles andere ist neu oder zumindest nicht mehr zu erkennen.

Der Compiler

Die wichtigste Veränderung, die der Compiler erfahren hat, ist die direkte Adressierung anstatt der PC-relativen Adressierung. Dadurch fällt die oft bemängelte Einschränkung der Länge eines Codesegments auf 32 KByte weg. Der Preis dafür sind längere Programme mit längeren Ausführungszeiten. Aber keine Angst, es geht dabei nicht um Faktoren, sondern um 5-10 Prozent.

Wie es sich für einen modernen C-Compiler gehört, kann Laser C auch Aufzählungstypen (Enumeration Types) und die Zuweisung von Strukturen (Structure Assignment). Letzteres war auch schon in Megamax C erlaubt. Der Datentyp int ist immer noch 2 Byte bzw. ein Wort lang. Eine Tatsache, die beim Portieren von Programmen aus der UNIX Welt immer zu Problemen führt.

Der Linker

Der Linker hat auch ein paar neue Features erhalten. Er ist in der Lage, Objekte im DRI-Format zu verarbeiten. Dadurch ist eine Verbindung zu anderen Programmiersprachen geschaffen (z.B. Pascal). Er kann außerdem eine Symboltabelle erzeugen, in das erzeugte Programm einbinden und ein Mapfile erzeugen. Beides erleichtert die Fehlersuche mit dem Debugger, auf den ich noch zu sprechen komme. Es besteht weiterhin die Möglichkeit ein Programm an eine frei wählbare Adresse zu linken. Für diese Option fällt mir in der normalen ATARI-Umgebung keine Anwendung ein. Man kann damit Programme für einen Einplatinen-Computer ohne Betriebssystem erstellen und den ST als Entwicklungsmaschine benutzen. Der gewöhnliche Benutzer wird diese Option aber nicht brauchen.

Will man ein Desk-Accessory linken, muß man nicht wie bisher eine Library zusätzlich angeben, sondern der Linker erkennt an der Extension '.ACC', daß es sich um ein Accessory handelt und bindet den erforderlichen Code automatisch mit ein.

Um eigene Objektbibliotheken (Libraries) zu erstellen, wird ein Archiver oder auch Librarian mitgeliefert. Die Ansteuerung Programms erfolgt über eine Kommandozeile und nicht interaktiv wie bei Megamax C. An dieser Stelle macht Laser C in punkto Benutzerfreundlichkeit einen klaren Rückschritt. Dafür kann man sowohl Laser C- als auch DRI-Bibliotheken verwalten.

Auch vor dem Disassembler wurde nicht halt gemacht. Er kann jetzt nicht nur Laser C, sondern auch DRI-Objektfiles und GEMDOS-Programme entschlüsseln. Dabei werden eventuell vorhandene Symbole verwendet oder neue generiert.

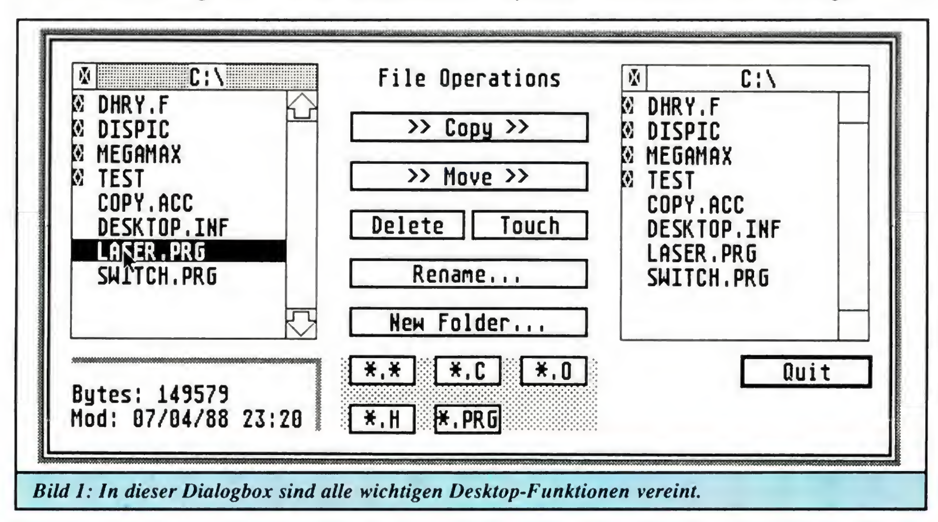


Bild 1: In dieser Dialogbox sind alle wichtigen Desktop-Funktionen vereint.

Neues Make-Utility

Das Make-Utility hat zu meiner persönlichen Freude einen erheblich gesteigerten Funktionsumfang erhalten. Offensichtlich wurde eine Anleihe bei UNIX gemacht. Es gibt Macros, implicit Macros, dynamic Dependencies und Transformation Rules. Es würde den Umfang dieses Beitrags sprengen, wenn ich an dieser Stelle die Funktionalität von Make erklären wollte. Aber lassen sich versichern, daß die Lektüre der 10 Seiten im Handbuch keine verlorene Zeit ist, auch wenn der Groschen nicht auf anhieb fällt.

	Compile+Link	Ausführung	Größe	Test
Laser C	5.3 s/4.8 s	2.7 s	12052 Byte	Sieb
Turbo C	12.3 s	2.8 s	3357 Byte	Sieb
Megamax	26.5 s	2.6 s	6955 Byte	Sieb
Laser C	18.1s/13.3s	845 D/s	13918 Byte	Dhrystone
Turbo C	20.0 s	1436 D/s	9108 Byte	Dhrystone
Megamax	45.1 s	920 D/s	9047 Byte	Dhrystone

Bild 2: Testergebnisse

Neues bei Laser C

Bis jetzt habe ich mich mit den Veränderungen an Programmteilen, die schon von Megamax C her bekannt sind beschäftigt. Nun möchte ich auf Neuheiten eingehen. Um fremde (oder auch eigene) Objektfiles zu erforschen, gibt es den Symbol Namer. Dieses Hilfsprogramm zeigt die Symbole in einem Objektfile im Laser C- oder DRI-Format oder im GEMDOS-Programmformat an. Ein weiteres Hilfsmittel ist Egrep (UNIX, ick hör dir trapsen), mit dem man einen Textstring in einer oder mehreren Dateien suchen lassen kann. Dabei sind sogenannte Wildcards erlaubt, die es ermöglichen, z.B. alle Kommentare aus einem C-Programm aufzulisten oder auch zu entfernen, sich die Namen aller in einem C-Programm definierten Funktionen ausgeben zu lassen usw.

Unter der Kategorie Neuheiten läuft auch die Shell von Laser C, die es wirklich in sich hat. Sie ist eine gelungene Mischung aus Kommando-Interpreter und GEM-Shell. Man kann die meisten Funktionen sowohl von einer Kommandozeile, als auch über die Menüleiste mit entsprechenden Dialogboxen aufrufen. Es ist also hinsichtlich der Benutzerschnittstelle für jeden Geschmack gesorgt. Das beste an der Shell ist aber der Cache. Das heißt, man kann Programme und Daten, nachdem sie erst einmal im Speicher sind, immer wieder verwenden, ohne sie erneut von der Disk zu laden. Die daraus resultierende Verarbeitungsgeschwindigkeit schlägt die Konkurrenz um Längen. In Bild 2 gibt eine kleine Tabelle die gemessenen Zeiten wieder. Bei der Tabelle handelt es sich um die gleichen Werte wie auf Seite 11 in der ST-Computer 6/88, erweitert um die Daten von Laser C. Bei Laser C sind jeweils zwei Werte für die Compile+Link-Zeit angegeben. Der erste Wert gibt die Zeit wieder, die beim ersten Durchlauf, also ohne Cache, und beim zweiten Durchlauf mit Cache gemessen wurden. Die Zeiten sind handgestoppt und wurden auf einem 520 ST+ mit einer SH 204 unter Blitter-TOS gemessen.

```
#include      <stdio.h>

long Fac(n)
long   n;
{
       if(n > 1) return(n * Fac(n-1));
       else return(n);
}

main()
{
       printf("Fakultaet von 7 = %ld",Fac(7L));
       getchar();
}
```

Bild 3: Beispiel für Rekursion

Zeit ist Geld

Wer mit einer Floppy arbeitet, kann sich bei Laser C die RAM-Disk sparen. Wenn erst einmal alles im Speicher ist, hat die Floppy nichts mehr zu tun. Man kann angeben, welche Programme gleich nach dem Start in den Speicher geladen und dort resident gemacht werden sollen. Die Quelltexte und Objektdateien werden automatisch in den Cache geladen und von dort aus verwendet. Die Cache-Logik ist der Schlüssel zu den superschnellen Turn-around-Zeiten von Laser C. Um Geschwindigkeit zu ereichen, muß man meistens Kompromisse eingehen, so auch bei dem Cache von Laser C. Das Programm ist nicht mehr unabhängig von der TOS Version. Es werden zwar alle bisher bekannten Versionen unterstützt, aber es kann mit neueren Versionen Probleme geben. Oliver Joppich von Application Systems /// Heidelberg versichert jedoch, das die Anpassung auf ein neues TOS im Configurationsfile der Shell erfolgen kann und kein teures Update erfordert.

Noch ein paar Worte zur Benutzeroberfläche der Shell. Wie schon erwähnt, gibt es einen Kommando-Interpreter, der in einem Fenster läuft. Wer UNIX oder ähnliches gewöhnt ist, fühlt sich dort schnell zu Hause, wenn auch nur ein kleiner Teil der UNIX-Tools zur Verfügung steht. Wer wie ich zu faul ist, sich Kommandos und Dateinamen zu merken, kann auch interaktiv die Kommandos, die man normalerweise im Desktop ausführt, verwenden. In Bild 1 ist die Dialogbox zu sehen, mit der man Dateien kopieren, löschen, umbenennen und verschieben kann. Die Auswahl in der linken Selectorbox ist dabei die Quelle und die rechte Selectorbox das Ziel. Klickt man auf eine Datei, wird in der linken, unteren Ecke die Länge sowie Zeit und Datum der letzten Veränderung angezeigt.

Minimake macht's möglich

Ähnlich sieht auch die Dialogbox aus, mit deren Hilfe man den Linker aufrufen kann. Man wählt aus einem Fileselector Objektfiles aus und hängt sie an die Liste der zu linkenden Objekte an. Diese Liste kann gespeichert und ähnlich wie ein Makefile verwendet werden. Ruft man den Menüpunkt 'Run' auf, so prüft die

```
>g
—> C:\FAC.C:main()
  —> C:\FAC.C:Fac(n = 7)
    —> C:\FAC.C:Fac(n = 6)
      —> C:\FAC.C:Fac(n = 5)
        —> C:\FAC.C:Fac(n = 4)
          —> C:\FAC.C:Fac(n = 3)
            —> C:\FAC.C:Fac(n = 2)
              —> C:\FAC.C:Fac(n = 1)
              <— \FAC.C:Fac(1)
            <— \FAC.C:Fac(2)
          <— \FAC.C:Fac(6)
        <— \FAC.C:Fac(18)
      <— \FAC.C:Fac(78)
    <— \FAC.C:Fac(2d0)
  <— \FAC.C:Fac(13b0)
<— \FAC.C:main(a)
Program Terminated.
> exit
```

Bild 4: Anzeige der Funktionsaufrufe im Debugger

Shell, ob die Objektfiles aktuell sind. Falls nicht, compiliert sie die entsprechenden Quelltexte und linkt dann die Applikation und startet sie. Man hat zwar nicht die volle Funktionalität von Make, aber für Programme, die sich aus nur wenigen Dateien zusammensetzen, reicht dieses Minimake völlig aus und hat den Vorteil, daß es fast keinen Aufwand erfordert. Die Objekte zum Linken muß man ja sowieso angeben, und ein zusätzlicher Klick zum Abspeichern erspart später viele Klicks zum Compilieren und Linken.

Die Shell enthält auch den Editor. An dieser Stelle ist es schwer ein objektives Urteil abzugeben. Ich bin von dem Editor begeistert. Mein Kollege Stefan Höhn, übrigens ein Tempus-Fan wie ich, war nicht sehr begeistert. Es ist also Geschmackssache. Wer den Artikel über Turbo-C, der auch aus meiner Feder stammt, gelesen hat, weiß, daß ich ein Freund des Apple Macintosh und der Entwicklungsumgebung MPW bin. Der Editor von Laser C bedient sich wie sonst üblicherweise ein Editor auf dem Mac. Möglicherweise ist das gewöhnungsbedürftig, aber ich glaube, man kann sich damit anfreunden. Der Funktionsumfang und die Geschwindigkeit ist für einen Quelltext-Editor ausreichend (ich gebe zu, Tempus ist schneller). Was mir nicht gefallen hat, ist die Tasache, daß nicht zeilenweise gescrollt wird. Wenn der Cursor am unteren Rand eines Textfensters angekommen ist und man drückt noch einmal die Cursordown-Taste, wird der Text um 5 Zeilen nach oben gescrollt, und der Cursor steht nicht mehr am Ende, sondern mitten im Fenster.

Der Debugger

Den wichtigsten Punkt habe ich mir für das Ende aufgehoben, nämlich den Debugger. Auch zu diesem Thema wurde ein eher unkonventioneller Ansatz gewählt. Der Debugger wird zu dem Benutzerprogramm gelinkt und ist dessen fester Bestandteil. Startet man sein Programm, meldet sich direkt der Debugger und lädt als erstes die Symboltabelle. Anschließend kann man Kommandos des Debuggers ausführen. Alle Kommandos verstehen Symbole. Man kann also Variablen oder Funktionen mit ihrem Namen ansprechen. Die Funktion 'list' zeigt Funktionen, Variablen, Breakpoints, Tracepoints und Memory Watches an. Breakpoints und Tracepoints setze ich als bekannt voraus. Was aber sind Memory Watches? Das sind Variablen, die der Debugger ständig überwacht, und falls sie ihren Wert ändern, bewirken, daß von der Applikation in den Debugger verzweigt wird. Diese Funktion ist sehr nützlich, wenn Variablen ohne ersichtlichen Grund plötzlich ihren Wert ändern. Wenn man mit mehreren Personen an einem Programm arbeitet, will es ja nie einer gewesen sein. Memory Watches helfen den Übeltäter dingfest zu machen. Leider muß ich anmerken, daß diese Funktion in der Testversion nicht einwandfrei funktionierte. Ich gehe aber davon aus, daß ein so wichtiges Feature in der endgültigen Version voll verfügbar ist.

Ein besonderer Clou des Debuggers ist die Erweiterbarkeit. Wer will kann den Debugger um zusätzliche Datentypen oder Funktionen erweitern. Die Schnittstelle ist offen gelegt und von C aus problemlos zu benutzen. Für dieses Schmankerl gebührt den Entwicklern ein dickes Lob. Offenheit ist Trumpf.

Ich habe ein kleines Programm geschrieben, um Ihnen die Arbeitsweise des Debuggers zu demonstrieren. Das Listing ist in Bild 3 abgedruckt. Das Programm berechnet rekursiv die Fakultät der Zahl 7. Bild 4 Zeigt die Ausgabe des Debuggers. Wie man sieht wird zu jedem Funktionsaufruf der Name des Sourcefiles und der Funktionsname mit den Übergabeparametern angezeigt. Die Schachtelungstiefe wird durch Einrücken kenntlich gemacht. An der Richtung des Pfeils vor der entsprechenden Zeile erkennt man, ob die Funktion aufgerufen oder verlassen wurde. Beim Verlassen wird anstelle der Übergabeparameter der Rückgabewert der Funktion ausgegeben.

Soviel zu der getesteten Version von Laser C. Es ist eine englische Testversion mit englischem Handbuch. Das Paperback-Handbuch beschreibt auf 640 Seiten alles, was man über Laser C und die Libraries wissen muß. Es gibt UNIX-kompatible Libraries und natürlich GEMDOS, BIOS, XBIOS, VDI und AES. Die Funktionen der einzelnen Libraries sind ausführlich beschrieben, oft sogar mit Beispielen vertieft.

Das ganze System wird als eingedeutschte Version (auch mit deutschem Handbuch) voraussichtlich zur ATARI Messe verfügbar sein. Der Preis soll 398 DM betragen. Wer schon stolzer Besitzer von Megamax C ist kann für 80 DM ein Update in Form von 2 Disketten und dem englischen Handbuch bekommen.

JL

Bezugsquelle:

Application Systems /// Heidelberg
Englerstraße 3
6900 Heidelberg
Tel.: 06221/300002

ENDE

MOUSI - The Switcher

Das ewige Umstecken hat ein Ende ! Mit MOUSI können Sie 2 Joysticks & 1 Maus gleichzeitig an Ihren ST anschließen, wobei für Port 2 eine Maus-Joystick-Umschaltung besteht. MOUSI wird über ein 50 cm langes Kabel an Ihren ST angeschlossen und kostet komplett mit Anleitung 99,- DM

ST-PC: ST-Spezialgehäuse ab 149,- DM eigens für 260/520 ST entwickelt. Platz für Hauptplatine, Netzteil, Floppy's, Harddisks.... Fertig bearbeitet !

Druckerständer aus Plexiglas 28,- DM

Außerdem sehr günstig: Uhren (29,-) // Zweitlaufwerke (299,-) // FUJI-DISKS (27,95) // BTX (248,-) // MIDI Hard & Software (Info 0,80) // Infos Gratis HÄNDLERANFRAGEN ERWÜNSCHT!

LACOM // Computer & Musik

Ilias Lazaridis - Emscherstr. 45 - 4200 Oberhausen 11

☎ 0208 / 65 06 69



SETH BEHLER DIPL.-SOZIOLOGE — **BÜRO f. SYSTEMBERATUNG & EVALUATION**

Tel. 0 26 66 - 16 37 · Heidestr. 12 · 5519 Freilingen

L.I.Z.A.-STATISTIK für ST unter GEM Verarbeitet Fremdformate (ADI, VIP, ASCII); bis zu 10 Variablen mit je 1000 Fällen parallel; Grafiken auf Drucker oder Disk (Piecharts, Histogramme, Kurven); Transformationen und Gruppenbildungen; Descriptive Statistik und Verteilungstests; Analyse von Varianz, Korrelation und Regression; Crosstables; parametrische und nichtparametrische Tests (CHI, U, T, Wilcoxon, F, Omnibus etc.); Verlaufs- & Ergebnisprotokoll auch auf Disk zur Einbindung in Textprogramme u.v.m.
L.I.Z.A. mit Handbuch und UP-Dates .... DM 196,-
Handbuch & Demodiskette (wird beim Kauf angerechnet) DM 30,-

L.I.Z.A.-TAFEL Deskaccessory zum Aufruf beliebiger statistischer Tafelwerte und Testgrößen aus laufenden Programmen – eine Ergänzung nicht nur für L.I.Z.A. DM 59,-

dBASE-ACCESS Deskaccessory zum direkten Zugriff auf dBASE-Datenbestände (Struktur, Anzeige, Suchen, Löschen, Ergänzen, Verändern) .... DM 59,-

PASCAL-TOOLS I: Datenbankzugriffe und statistische Routinen I – eine Sammlung linkfähiger Module und Assembler-Routinen für ST Pascal+
Library und Anleitung ohne Sourcecode ..... DM 89,-
mit Sourcecode .... DM 119,-

Infos gegen Rückporto – Preise zzgl. Versandkosten

Mit Modula ans GEM

Karl Vogel
GEM Programmierung in Modula-2 auf ATARI ST/Mega ST
AXIS

Karl Vogel: GEM Programmierung in Modula-2 auf ATARI ST/Mega ST.
1988, 120 S., kart.,
DM 25,– / sFr. 21,–
ISBN 3-905122-00-6

Das praxisorientierte Buch vermittelt Schritt für Schritt die Grundlagen der Programmierung GEM-typischer Elemente: Rolladenmenüs, editierbare Dialogboxen, manipulierbare Fenster. Es eignet sich sowohl zum Selbstlernen wie auch als Lehrmittel für Schulen und EDV-Kurse. Ausgehend von präzisen Aufgabestellungen werden auch charakteristische Demonstrationsprogramme entwickelt.

Bestellung an:

AXIS-Verlag
Postfach 522 · CH-8590 Romanshorn 3

EASY RIDER VERSION 1.1 IST DA!

EASY RIDER ist ein intelligenter Reassembler-EASY RIDER erzeugt von jedem Programm einen assemblierfähigen Quelltext!!-Datenbereiche werden von EASY RIDER mit großer Sicherheit erkannt! In der neuen Version können Datenbereiche zusätzlich manuell definiert werden!!-EASY RIDER wurde mit weiteren Suchfunktionen ausgestattet. Die Ausgabegeschwindigkeit wurde noch erheblich gesteigert.- EASY RIDER ermöglicht die Beseitigung von Fehlern in fremden Programmen, genauso wie die Einbindung von Assemblerroutinen in eigenen Programmen.- Wie dies am einfachsten zu bewerkstelligen ist, wird anhand von Beispielen und in dem neuen Handbuch ausführlich erläutert!! EASY RIDER erhalten Sie für 159,- DM + 4,70 DM Versandkosten von:

ANDREAS BORCHARD ☎ 0541/87024
WIESENBACHSTR. 2A **4500 OSNABRÜCK**

COMPY SHOP

Wir haben die neueste Software für Ihren ATARI ST!

Der neue Archimedes ist da.

Fordern Sie Informationen an.

Und vieles mehr! Fordern Sie unsere kostenlose ST-Software-Liste an!

COMPY-SHOP
Gneisenaustr. 29 · 4330 Mülheim Ruhr
Telefon (02 08) 49 71 69

wissenschaftliche
STATISTIK
mit
WiSTat

einfach in der Anwendung
vielseitig in der Leistung

- alle einfachen Testverfahren (t-Tests usw.)
- mehrfaktorielle Varianzanalysen
- Korrelationen, multiple Regression
- Faktoren-, Cluster-, Regressionsanalyse
- Item- und Diskriminanzanalyse
- Bequemes Dateienhandling
- Umfangreiche Datentransformationen
- Ausgabe aller Ergebnisse auch auf Drucker
- Hotline, Update Service, Sonderwünsche...

Wenn Sie ein besseres Statistikprogramm finden, dann kaufen Sie es!
Händleranfragen erwünscht!
mit Lehr-Handbuch nur 398,— DM
12-seitiges Info: **Thomas Leschner**
Universitätsstraße 40 · 3550 Marburg

Public-Domain-Software
»»» zu steinharten Preisen!!«««

Sie kennen uns noch nicht?
Dann nutzen Sie unser PD-Werbeangebot:
4 Atari-ST-Public-Domain-Disketten
gefüllt mit tollen Spielprogrammen, Utilities und Anwendersoftware – im Spezialverfahren kompaktiert auf einer 2-seitigen 3 1/2"-Diskette, dazu unsere ausführliche PD-Liste. ...alles zusammen bekommen Sie gegen Einsendung von
NUR DM 5,–
in Briefmarken
Scheck oder Bar!



PD-Liste gratis!

1-seitige Disketten nicht mehr lieferbar!

Hotline:
Tel.: 07 81 / 5 83 45

FsKS Ludwig Abt. Atari Kastanienallee 24 D-7600 Offenburg

OmegaSoft-Pascal,
das Pascal für Anspruchsvolle!
DM 445,—
Bericht siehe ST Dezember '87.

OS9 auf dem ST,
das Multiuser-, Multitasking-System.
Fragen Sie unseren Preis!
OS9-Software verschiedener Art.
Information anfordern.

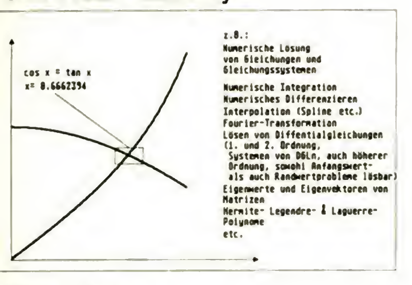


BYTE STUDIO BORKEN
Butenwall 14 · 4280 Borken Tel. (0 28 61) 21 47

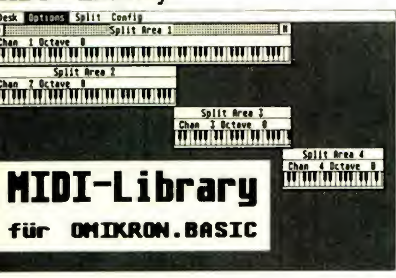
TATISTIK - Library

Umsatz 87/88

UMERIK - Library



IIDI - Library



IASKENEDITOR



.R.A.W.!





OMIKRON.BASIC: das Entwicklungssystem

„Der OMIKRON.BASIC-Interpreter stellt zusammen mit dem Compiler das beste BASIC-Entwicklungssystem für den Atari ST dar."
(ST Computer 9/87, S. 64)

Der Interpreter

Ein umfangreicher Befehlssatz, ein leistungsstarker Editor und höchste Ausführungsgeschwindigkeit kennzeichnen den OMIKRON.Interpreter.

- 19 Stellen Genauigkeit bei allen Funktionen
- Matrizen-Algebra
- Masken-INPUT, SORT-Befehl (mit deutschen Umlauten)
- Strukturiertes Programmieren: rekursive Prozeduren, mehrzeilige Funktionen
- 99 % MBASIC-kompatibel

OMIKRON.BASIC auf Modul **DM 229,–**

* Omikron Basic auf Diskette ist neuer Basic Standard auf Atari ST und liegt seit dem 1.07.88 jedem ST schon ab Werk bei.

Der Compiler

Noch mehr Tempo erreichen Ihre in OMIKRON.BASIC geschriebenen Programme durch unseren OMIKRON.Compiler.
Der Compiler erzeugt ein Assemblerprogramm (echte Maschinensprache, keinen P-Code!), das direkt vom Desktop aus angeklickt werden kann. Auch Accessories können Sie mit dem OMIKRON.Compiler erstellen. Durch CUTLIB werden nur diejenigen Ausführungsroutinen eingebunden, die tatsächlich gebraucht werden; die Folge sind außerordentlich kompakte Programme. Die Ablaufgeschwindigkeit der compilierten Programme ist außergewöhnlich. Der Tester in der mc meinte gar: **„Nur Assembler ist schneller"** (mc 2/87, S. 122)

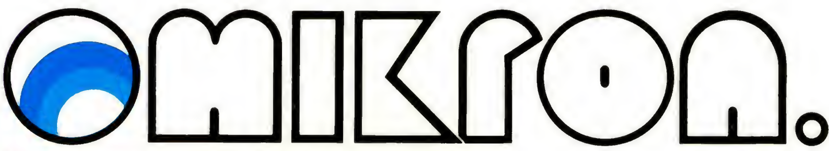
OMIKRON.Compiler **DM 179,–**
OMIKRON.Compiler für 68881
(vollkompatibel zum normalen Compiler) **DM 229,–**

Die Libraries

Wer professionell Programme entwickelt, möchte nicht jedesmal das Rad neu erfinden. Mit neuen Libraries (Befehls-Bibliotheken) und Hilfsprogrammen macht OMIKRON.BASIC Ihnen das Leben jetzt noch etwas leichter. Libraries erweitern den Befehlssatz: Sie laden einfach die Library, und die neuen Befehle stehen Ihnen sofort zur Verfügung.
Außer den links abgebildeten Programmen gibt's noch eine Robot-Library (ab Mai) und einen **I**ntegrierten **D**ebugger **E**ditor **A**ssembler **L**inker (kurz **IDEAL**), mit dem Sie Maschinenprogramme erstellen und austesten können (ST Magazin 19, Seite 74: „Debugger der Superklasse")

Alle links abgebildeten Programme **je DM 79,–**
IDEAL **DM 99,–**

Gerne schicken wir Ihnen nähere Informationen zu – Anruf genügt!



Programme für Programmierer

OMIKRON.SOFTWARE

Erlachstraße 15 · D-7534 Birkenfeld · ☎ (0 70 82) 53 86

Frankreich: OMIKRON · France, 11 Rue Dérodé, F-51100 Reims
Luxemburg: ELECOMP, 11 Avenue de la gare, L-4131 Esch-Alzette
Niederlande: Terminal Software Publicaties, Postbus 111, NL-5110 Baarle-Nassau
Österreich: Ueberreuter Media, Laudongasse 29, A-1082 Wien
Schweiz: Thali AG, Industriestrasse 6, CH-6285 Hitzkirch

Immer up to date

Mit dieser Sparte wollen wir allen unseren Lesern die Möglichkeit geben, sich über die neuesten Programm-Versionen zu informieren. Angegeben werden die aktuelle Versionsnummer, ein eventueller Kopierschutz, die Bildschirmauflösungen und der Speicherbedarf. Softwarefirmen ist es somit möglich, die ST-Computer-Leser über ihre Updates zu informieren.

Programmname	Version	Daten	Programmnam	Version	Daten
Adress ST / Check ST	1.0	N H	Link_it GFA	1.1	N HML
AnsiTerm	1.4	N	Link_it Omikron	2.0	N HML
1st BASIC Tool	1.1	N HML	Lisp Complete	1.01	N HM
Binlook	1.0	N HML	Lock_it	1.0	J HML
BTX für ST	1.0	N H 1M	Megamax C	1.1	N HML
Crypt_it	1.0	J HML	Mega Paint	1.0	N H 1M
Flash-Cache/Flash-Bak	1.0	N HM	Megamax Modula 2	3.5	N HM
Flexdisk	1.2	N HML	Micro C-Shell	2.70	N HM
1st_Freezer	1.0	N HML 1M	MT C-Shell	1.20	N HM 1M
GFA-Artist	1.0	N L	Multi ST	1.0	N HML 1M
GFA-BASIC 68881	1.3	N HML	Musix32	1.01	J H
GFA-Compiler	2.02	N HML	Omikron BASIC Compiler	2.4	N HML
GFA-Draft	2.1	N	Omikron BASIC Interpreter	3.0	N HML
GFA-Draft plus	1.5	N	PAM's TERM/4014	3.012	N H
GFA-Interpreter	2.02	N HML	PAM's TurboDisk	1.7	N HML
GFA-Objekt	1.1	N HM	PAM's NET	1.0	N HML
GFA-Starter	1.1	N HML	PC ditto Euroversion	3.64	N HML
GFA-Vektor	1.0	N	Pro Sound Designer	1.2	L
Hard Disk Accelerator	1.0	N HML	Pro Sprite Designer	1.0	L
Hard Disk Toolkit	1.05	N HM	Search!	1.31	N HM
Harddisk Utility	2.0	N HM	Signum! zwei	1.0	N H
Imagic	1.0	N HML	1st_Speeder	1.01	N HML
Intelligent Spooler	1.01	N HML	1st_Speeder 2	1.0	N HML 1M
K-Resource	1.1	N HM	STAD	1.3	N H
Label ST	1.0	N HML	ST Pascal plus	2.02	N HM
Laser Deluxe	1.0	N HML 1M	Tempus	2.0	N HM
1st_Lektor	1.2	HM	Transfile ST plus	2.0	N HM
Lern ST	1.22	N HML	VSH Manager	1.11	N HML 1M

Irrtum vorbehalten

Daten-Legende : N = kein Kopierschutz, J = Kopierschutz, H = hohe Auflösung, M = mittlere Auflösung
L = niedrige Auflösung, 1M = mindestens 1 Megabyte Speicher notwendig

Pst...
Glauben Sie, daß der
Heim Verlag auf der
ATARI-Messe in Düsseldorf
die ST-Buchidee des Jahres
enthüllt? Bitte umblättern!



ATARI ST AUFRÜSTUNGEN

Wir liefern alle hardwaremäßigen Voraussetzungen, so z. B.:

SPEICHERERWEITERUNGSKARTEN
passend für alle ST-Typen
- sehr leichter Einbau **ohne** löten durch ausführlich bebilderte Einbauanleitung
- kostenloser Einbau durch uns möglich
- vergoldete Microsteckkontakte ergeben optimalen Kontakt und flimmerfreien Bildschirm
- keine zusätzliche Software erforderlich (TOS erkennt die Erweiterung)
- mit 256K- oder 1Mbit-Chip bis 4MByte (auch nachträglich) bestückbar
- **1 Jahr Garantie**
- **Option:** batteriegepufferte Echtzeituhr auf der Erweiterungsplatine incl. Software

Preis: ab **DM 261,00** incl. RAMDISK.Programm und MS-DOS-Format sowie weitere nützliche Programme

weitere Produkte
Festplatten: von Markenherstellern ATARI, VORTEX, SUPRA
20 – 250 MByte ab DM **1.098**
Neu: Harddisk **40 MByte** zum **direkten Einbau** in das **Mega ST Rechnergehäuse!!!**

HOSTADAPTER:
HST/S SUPRA DM **398**
SCSI-Hostadapter SUPRA liefert vollständigen SCSI-Bus: für alle Harddisks u. ä. mit SCSI-Controller (8 LW) incl. 1a-Software.
HST/PC BERKLEY DM **548**
SCSI Host, incl. Controller zum Anschl. v. bis 2 IBM-PC komp. Harddisk (MFM) incl. hervorrragender Software

48 STUNDEN REPARATURSERVICE
Unsere Produkte erhalten Sie direkt von uns oder im Fachhandel.
Weitere Infos erhalten Sie gerne von uns

CSH INGENIEURBÜRO Dipl.-Ing. M. H. Krompasky
Schillerring 19 · 8751 Grosswallstadt · Tel. 0 60 22/2 44 05

Lattice C V3.04 .... DM 194,95

MCC-Assembler V11.1	DM 99,95	MCC-Pascal V2.02	DM 175,00
Cambridge Lisp	DM 299,00	BCPL	DM 189,00
Copy II/ST V2.5	DM 79,00	GST-Makro-Assembler	DM 77,95
Psion Chess deutsch	DM 59,95	GATO-U-Boot-Simulat.	DM 60,95
Defender of the Crown	DM 59,95	Barbarian	DM 55,95
Modula II V3.01 Stand.	DM 189,00	Time Bandits	DM 49,95
Electornic Pool	DM 42,95	Shuttle II	DM 52,95
PTX-Term ST/Akustkk.	DM 198,00	BTX-Term ST/Postm.	DM 249,00
Public Domain ab 1 St.	DM 4,00	No-Name MF2DD 2s.	DM 2,29
Einzellaufw. NEC1037A	DM 339,00	Flightsimulator II	DM 89,00
Megamax Modula II dt.	DM 319,00	Megamax C dt.	DM 319,00
Signum!Zwei dt.	DM 369,00	STAD dt.	DM 149,00

Kostenlose Prospekte auch für IBM & Amiga
Betriebsurlaub vom 12.8. – 4.9.88

CWTG
Computerversand CWTG Joachim Tiede
Bergstraße 13 ★★★ 7109 Roigheim
Tel./BTX 0 62 98/30 98 von 17-19 Uhr
SCHRIFTLICHE HÄNDLERANFRAGEN ERWÜNSCHT

Ecosoft Economy Software AG
Postfach 30, 7701 Büsingen, Tel. 07734 - 2742

Software (fast) gratis

- **Größtes Angebot von Prüf-Software und Frei-Programmen Europas:** Über 3'000 Disketten für IBM-PC/ Kompatible, Macintosh, Amiga, Atari ST, C64/128, Apple II.
- Viele **deutsche Programme**, speziell für Firmen, Selbständige, Privatpersonen, Schulen.
- **Software gratis**, Sie bezahlen nur eine Vermittlungsgebühr von DM 14.40 oder weniger je Diskette.
- HOTLINE: Tel. Anwenderunterstützung (3 Techniker von 8-18 Uhr).

Verzeichnis gratis

Bitte Computermodell angeben. Gegen Einsendung dieses Inserates erhalten Sie zusätzlich einen Gutschein für eine Gratis-Diskette. 483

Computertechnik Z. Zaporowski
Dreieckstraße 2b, 5800 Hagen1, Tel. 02331 / 86555

Neu:
Die Monitorumschaltung jetzt in drei Ausführungen schon ab **DM 29.90**
1. Bausatz **DM 29.90**
2. Fertiggerät **DM 44.90**
3. Fertiggerät mit einem zusätzlichen Monochrom-Video-Signal (BAS), auch geeignet fur den Anschluß eines Videorecorders **DM 59.90**

Das PAL-Interface:
Jetzt ebenfalls in drei Ausführungen lieferbar:

PAL-Interface FBAS **DM 159,-**
Fertiggerät mit FBAS-Wandler zum Anschluß an Geräte mit FBAS oder AV-Buchse.

PAL-Interface III **DM 198,-**
Fertiggerät mit Modulator zum Anschluß an jeden handelsüblichen Fernseher.

PAL-Interface III-2 **DM 239,-**
Fertiggerät mit integriertem Monitorumschalter und Modulator.

Das Uhr Modul
Datum und Uhrzeit immer aktuell. Das Modul wird einfach in den ROM-Port gesteckt, daher kein Garantieverlust des Rechners. Mit Bedienungsanleitung und Treibersoftware auf Diskette. **DM 69,00**

Versandkostenpauschale DM 8,50
Transportversicherung: DM 2,-
Ausland: Versandkosten DM 15,-, nur Vorkasse

ST-Gratisinfo anfordern!

Btx-Manager
Atari ST / Mega ST + Btx
Die intelligente Komplettlösung...



Btx-Manager V3.0
Softwaredecoder anschlußfertig mit Btx-Interface an Postbox **428 DM**
Btx-Manager V3.0
an Ak.koppler **328 DM**
Btx-Interface
Anschlußdirekt an Btx-Postbox **128 DM**
Telex-Manager
mit Menü-Manager **198 DM**
Menü-Manager **98 DM**
Infos anfordern!

Drews EDV + Btx **6900 Heidelberg**
Tel. 0 62 21 - 2 99 00 o. 16 33 23 **Bergheimer Str. 134 b**
Btx ★29900#
Atari ST / Mega ST sind eingetr. Warenz. der Atari Corp.

Ich bestelle:

_______ Stück „BUCH '88" (Arbeitstitel), das Buch für jeden ST-Anwender, à DM 59,— zuzüglich DM 5,— Versandkosten (unabhängig von der bestellten Stückzahl).

☐ per Nachnahme
☐ Verrechnungsscheck liegt bei

____
Name, Vorname

____
Straße, Hausnummer

____
PLZ, Ort

Heim Verlag
Heidelberger Landstr. 194
6100 Darmstadt 13

Pst... Hier sind wir.

Sollten Sie nicht nach Düsseldorf kommen können, bestellen Sie „BUCH '88" (der Buchtitel kann erst auf der Messe bekanntgegeben werden) einfach per Coupon.

ST-COMPUTER PUBLIC DOMAIN

Das Sommerloch

Liebe Leser,

wenn der Sommer uns mit seinem regnerischen Wetter beglückt, bringt es eines mit sich: das Sommerloch. Es geht das Gerücht, daß gerade in der Computerbranche, im Sommer nichts läuft. 'Viele fahren in den Urlaub und lassen den Computer links liegen'. Doch auch in diesem Monat haben wieder viele fleißige Programmierer ihre Software der Sammlung zur Verfügung gestellt. Wir danken für die Beteiligung.

Ihre Merlin-Computer

Neu in diesem Monat!



DRUCKUTILITIES

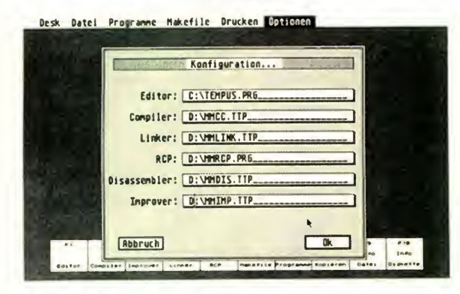


DRUCKERTEST: Das im letzten Heft beschriebene Druckertestprogramm. Es ermittelt für jeden Drucker die wahren Zeiten für Text- und Grafikdruck. Testen sie Ihren Drucker.

CFG-ANALYSE: Wandelt 1st_Word CFG-File in edierbare HEX-Datei zurück.



SHELLS



MEGAMAX_SHELL: Komfortable Shell. Direkter Aufruf von Editor, Compiler, Linker, Improver, Disassambler, RCP und anderen Programmen. Eingebautes MAKE. Druckerausgabe mit Anpassung. Zahlreiche Einstellungen.

LATTICE_SHELL: Shell für LATTICE C. Im Stil von Turbopascal.

XLISP_SHELL: Leicht zu bedienende Shell vereinfacht die Bedienung von XLISP (PD 7), (mit Source-Code in C).

PROLOG_SHELL: Shell für Prolog 10 von PD 11. Aufruf von Prolog, Editor und Optionen per Icon-Klick (mit Source in C).



CHEMIEPROGRAMME



LABORANT: Programm für den angehenden Chemiker (Mittel- bis Oberstufe, Studienbeginn). Molmasse, Mengenanteile, Gleichungsdurchrechnung, Empirische Formel, Titrationsauswertung, Umrechnungen (Mol, Menge, Lösungen, Konzentration, Volumen,...), Fehlerrechnung, Regression, Interpolation. Identifizierung von ca. 90% der anorganischen Verbindungen, Überprüfung auf Korrektheit anhand der Wertigkeit. (s/w)

ANALYSE: Hilfe für Chemiker, CTA's und Laboranten bei Elemantaranalysen. Überprüft, ob Meßwerte ein sinnvolles Element ergibt oder ob es doch ein anderes, verunreinigtes sein könnte. Detaillierte Anleitung. Beherrscht alle Elemente des Periodensystems. (altes TOS oder Disk-TOS, s/w)

MS_PLOT: Programm zum Zeichnen von Massenspektren in der Chemie. (s/w)



ASTROPHYSIK



SWING_BY: Simulationsspiel mit der Schwerkraft. In drei verschiedenen Variationen müssen Sie versuchen einen Planeten an einigen Sonnen vorbei zu manövrieren und an sein Ziel zu bringen, ähnlich den Sateliten, die aus unserem Sonnensystem befördert werden. (mit Source in GFA-Basic). (s/w)

PLANET: Berechnung des Sternenhimmels zu verschiedenen Zeiten und an verschiedenen Orten. Stellung der Planeten. Blickrichtung: Himmel, Horizont, Nord-/Süd-Karten.

STERNZEIT: Ähnlich PLANET. Erkennt Sternbilder und Sternnamen. (mit Source in GFA-Basic). (s/w)

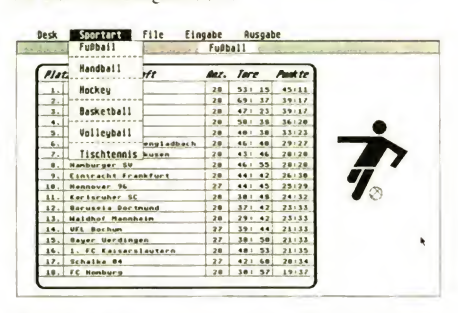


BENZIN UND SPORT



BENZIN: Benzinkostenrechnung, grafische Auswertung, Listendruck, Statistik.

KFZ: KFZ-Statistik. Benzinkostenrechnung von bis zu 4 Fahrzeugen. (s/w)



LIGA_TAB: Tabellenberechnung für Vereine, Spieler oder Fans. Saisonübersicht, Vereinsstatistik, aktuelle Tabelle, Spielergebnisse in folgenden Sportarten: Fußball, Handball, Hockey, Basketball, Volleyball und Tischtennis. (s/w)

TABELLEN: Tabellenverwaltung für Sportarten aller Art.



DATENBANKEN



LITERATUR: Verwaltungsprogramm für Zeitschriftenartikel. Platz für umfangreiche Kommentare. Suche nach Bruchstücken.

P_A_D_M: Professional Atari Data Manager. dBASE II-ähnliche Datenbank mit Kommandosprache. Flexible Datenverwaltung. Programmierung mit ca. 60 Befehlen.

BÖRSEN: Peters Börsen Manager. Für alle, die Geld an der Börse verdienen wollen und dazu eine Übersicht brauchen oder für die, die einfach einmal reinschnuppern wollen. Grafische Auswertung

ST-COMPUTER PUBLIC DOMAIN



SPIELE



HASE UND IGEL: Wunderschön gestaltetes Brettspiel. Hase und Igel auf dem Wettlauf ins Ziel. Unterwegs müssen Sie Karotten und Salate fressen aber nicht überhastet agieren.
Spieler: 2 Menschen und ein Computer. (s/w)

DODG_EM: Zwei Rennwagen auf Kollisionskurs. Nur durch schnelle Reaktion kann man ausweichen. Mit gesampeltem Sound. Einfache Grafik aber interessante Spielidee. (s/w)



DIVERSES

MORSE: Morse-Übungsprogramm. Gehörtraining durch verschieden schnell gemorste Zeichen. Fünfergruppen, Einzelzeichenabfrage, Alphabettraining.

CW_TRAINER: Morsetrainer mit realistischen Störgeräuschen. Ausgabe von Zeichen in verschiedenen Geschwindigkeiten.
Programm übersetzt Ihre Morsezeichen, die über die Maus eingemorst werden. (s/w)

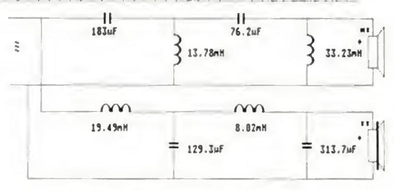
STUNDENPLAN: Dient zum Ausdruck eines Stundenplans. (s/w)

TOTOSYSTEM: Errechnet aus drei vorgegebenen Bänken ein Toto-System mit Gewinngarantie. Bedruckt sogar Totoscheine mit den berechneten Daten.

WIDERSTAND: Gibt nach Anklicken der Farbkombination den Widerstandswert aus.

EISENBAHN: Programm kontrolliert die Weichensteuerung eines zweiweichigen Bahnhofsbereiches. Computersteuerung über Relais und Fahrtstreckenmelder (z.B. Lichtschranken) über Joystickport realisiert. Zug- und Signalstandanzeige am Bildschirm. (s/w)

LAUTSPRECHER: Berechnung von Frequenzfiltern und Gehäusevolumen(Bassreflex) für Zweiwegsysteme. Butterworth und Besselfilter erster bis fünfter Ordnung.



Wir verwenden ausschließlich doppelseitige Markendisketten (2DD) der Firmen Maxell und JVC.
Sämtliche Disketten sind geprüft, fehlerfrei und virenlos.



Auf der nächsten Seite finden Sie weitere Disketten aus unserer Sammlung

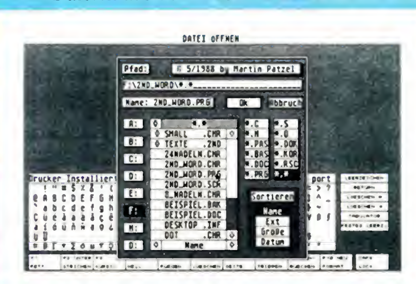
Updates

Viele PD-Autoren erweitern oder verbessern ihre Programme aufgrund der Resonanz aus der Leserschaft. So auch in diesem Monat. Auf den Original-Disketten befinden sich sodann die neuesten Versionen.

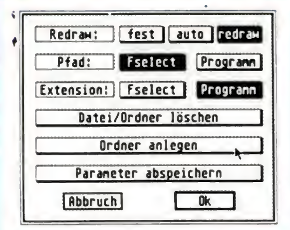
Folgende Disketten wurden aktualisiert:

21 HAUSHALT: Verbesserte Version.
25 MÜHLE: Mühle als Accessoiry. Jetzt auch mit Blitter-TOS.
32 AGO-FINANZMATHEMATIK: Jetzt an Blitter-TOS angepaßt. Umfangreiche Erweiterungen im Bereich der Rentenberechnung, (s/w).
58 DISK_LABEL: Ändern des Diskettennamens. Jetzt auch für Blitter-TOS.
69 ASTROLABIUM: Zusätzlich Sternnamen, Sternbilder, neue Grafikmöglichkeiten, 9000 Sterne, (s/w).
91 PAUK: Neue Version des Pauker-Programms.
126 OR-SPREAD: Erweiterte Version: Grössere Felder (500*500), absturzsicher, (s/w).
141 MONOMON: Verbesserte Version des Diskmonitor. Jetzt auch für Hyperformat-Disketten, sauberer Ramdisk-Zugriff, geänderte Druckroutine, (s/w).
142 SCANNER: Erweitere Version der Scannersoftware des Projektes aus ST-Computer 4/88.
123 FILESELECT: Wie Sie im Bild sehen können hat sich schon optisch einiges geändert. Aber auch programmtechnisch wurden viele, hauptsächlich aus dem Leserkreis entstandene, Verbesserungsvorschläge berücksichtigt. Diese wären in Kürze:

- Retten des Bildhintergrundes (läuft nun problemlos mit STAD)
- Repeat in den Rollbalken
- Pfad- und Extentionsteuerung vom Programm oder der Box
- Löschen und Anlegen von Ordnern und Dateien aus dem Fileselector heraus.



Die neue FILESELECTOR-BOX



Durch Anklicken der ersten Zeile kommt man in dieses Options-Menü

Es gelten die normalen Versandbedingungen (siehe rechte Spalte)

BITTE BEACHTEN

Sämtliche Disketten können ab dem Erstverkaufstag der ST-Computer direkt bei der MERLIN-Computer bezogen werden. Wir haben für Sie den schnellstmöglichen Versandservice eingerichtet. Lieferung innerhalb einer Woche.
Bitte beachten Sie folgende Punkte:

1. Schriftliche Bestellung

- Der Unkostenbeitrag für eine Diskette beträgt DM 10,-
- Hinzu kommen Versandkosten von DM 5,- (Ausland DM 10.-)
- Bezahlung nur per Scheck oder Nachnahme
- (Im Ausland nur Vorrauskasse möglich)
- **Ab 6 Disketten entfallen die Versandkosten (DM 5.- bzw. DM 10.-)**
- Bei Nachnahme zuzüglich DM 3,70 Nachnahmegebühr
- Bitte fügen Sie keine anderen Bestellungen oder Anfragen bei.

Adresse:
MERLIN'-Computer GmbH
´PD ST-Computer´
Postfach 5969
D-6236 Eschborn

2. Anruf genügt

'MERLIN'-Computer GmbH
´PD-Versand`
Tel.: 0 61 96 / 48 18 11
Mo-Fr 9⁰⁰ - 13⁰⁰ und 14⁰⁰ -17⁰⁰ Uhr

- Nur gegen Nachnahme (Gebühr DM 3,70)

Bei Fragen bezüglich der Programme stehen wir Ihnen gerne zur Verfügung.

ST-COMPUTER PUBLIC DOMAIN

Altbewährtes

Die nicht aufgeführten Disketten sind natürlich auch weiterhin erhältlich, nur wurde uns der Platz zu klein. Schauen sie dazu bitte in einer der vorherigen Ausgaben nach oder fordern Sie die Liste an (frank. Rückumschlag).

Diskette 7
-**XLISP 1.71**: (die Sprache der künstliche Intelligenz) incl. 54 KByte Dokumentation, Anleitung und Beispielen.

Diskette 11
-**PROLOG 10**: Die Sprache der 5. Genertion. Unterstützt GEM.
incl. 95 KByte Anleitung und Beispiele.

Diskette 23
-**DGDB**: Das große deutsche Ballerspiel: besonderes Aktionspiel (f)

Diskette 34
-**ASSEMBLERPAKET**: nach M68000-Standard (2-Pass Assembler. Disassembler, Debugger, Editor)

Diskette 40
-**DELUXE FONTMASTER ST**: Druckprogramm für Karten oder schönen Tabellen. Viele Zeichensätze, Proportionalschrift und Blocksatz, eigener Zeichensatzeditor.

Diskette 75
Randvoll mit Zeichensätzen für Fontmaster (Disk 40).

Diskette 49, 50, 51
-**VOLKSFORTH 83**: Die drei Disketten beinhalten ein komplettes, leistungsstarkes Forth-System für den ST. Alle Diskette werden benötigt.

Diskette 52
-**MOLEKÜL**: Darstellung chemischer Moleküle. In s/w und Farbe wobei in Farbe mit 3D-Effekt. Atom, Ionen oder Van der Waalsradius. Ein Molekül kann gedreht und sogar animiert werden.

Diskette 70
-**ORBIT**: Satellitenbahnberechnung. Animation über der Weltkarte (s/w)
-**HIGH END 2.2**: Programm zum Berechnen von Lautsprecherboxen aller Art. Lautsprecher und Boxenmaße. Datei für Speaker enthalten. (s/w)

Diskette 77
Utilities
-**FCOPY 2.0**: Schnelles Kopierprogramm. Formatieren, Verify, MultipleCopy, Scan und Teilkopie.
-**SPEEDER**: Verdoppelt Schreib- und Lesegeschwindigkeit.
-**FILECOPY**: Kopieren von Einzeldateien. Auch mit einer Floppy.
-**FORMATER**: Erhöht die Diskkapazität
-**FASTFORMAT**: Formatiert Schnell-ladedisks.

Diskette 88
Druckertreiber
-**NEC-EMU**: Hardcopytreiber für NEC P5/P6/P7. Erzeugt Hardcopies in 12 verschiedenen Größen und versch. Kontrasten. Enthält Druckerspooler.
-**LQ 800**: Hardcopytreiber für EPSON 24 Nadeldrucker in drei verschiedene Größen.
-**NEC_CODE**: Ein Programm zur Einstellung des NEC P6 /P7 u.ä..
-**SIMPLE**: Komfortables Hardcopy-Programm mit vielen Extras.
-**STX-80**: Hardcopytreiber für diesen Thermodrucker.
-**WORDPLUSTREIBER**: Druckertreibersammlung für folgende Drucker: NL10, FX1000, SP1000I.

Diskette 90
Utilities
-**MAXIDISK 4**: Die komprimierende Maxidisk mit max. 4MByte Kapazität. Läuft auf ALLEN TOS-Versionen. Komprimiert automatisch ihren Inhalt. RESETFEST.
-**TOPS**: The Other Pascal Shell (für PascalPlus). TEMPUS-fähig, erzeugt Crossreferenz. Kopieren, Löschen und Umbenennen von Files.
-**SIGNUMSHELL**: Eine Shell zu Signum. Umschaltung zwischen Signum Editor, den verschiedenen Druckprogrammen, den Zeicheneditoren per Menüleiste. Automatisches Install.
-**TEMPELMON 1.8**: DER weitverbreitete Monitor.
-**RAMDISK**: Gleiches Programm wie MAXIDISK, nur ohne Komprimiealgorithmus.
-**COPY**: Beide Ramdisk enthalten ein Autocopyprogramm zum Opieren bestimmter Files in Ramdisk.
-**RCS.WANDEL**: Wandelt "*.h-Files' nach Basic und Assembler.

Diskette 93
-**ADR_2**: Semiprofessionelles Adressverwaltungsprogramm mit Serienbrieferstellung (mit und ohne 1st Word). Edierbare Druckeranpassung (Steuerzeichen, Zeichenwandlung). erzeugt Adress- Telefon und Geburtstagslisten. Druckt Labels, Etiketten nach freidefinierbarem Format. 1st/2nd_Word-kompatibel (s/w)
-**DAT_TEXT**: Adressverwaltung für Mehrpersonenzugriff auf eine Datei. Datenzugriff nur durch Passwort. (s/w)

Diskette 100
-**ST-SPEECH**: Sprachausgabe auf Softwarebasis. Wandelt englischsprachige Texte automatisch in verständliche Lautschrift. (s/w)
-**FORTRAN SHELL**: Eine Shell für Fortran 77. Bequemes Aufrufen von Compiler, Linker, u.a. Programmen.
-**LIFE IS LIFE**: Life-Simulationsprogramm. (s/w)
-**FASTLIFE**: Life-Simulation in Volksforth (PD 49,50,51). Wahnsinnige Geschwindigkeit. (s/w)
-**GEMFRAC**: Berechnung fraktaler Landschaften in Farbe und s/w.
-**SHARP**: Basic Programme für Sharp-Basicrechner. Geeignet für Direktüberspielung mit Interface.

Diskette 116
-**µEMACS 3.8**: Editor. Deutsche Version. Eigene Macrosprache, Textverschlüsselung, Mailmerge, mehrere Screens, u.v.a.m.

Diskette 117
Terminalprogramm
-**UNITERM V 2.0**: Exclusives Terminalprogramm. VT200. VT102, VT100, VT52, 4010, u.a. Softscroll, Grafikübertragung, Funktionstastenbelegung...

Diskette 120
-**WISSEN SIE ES?**: Quizspiel mit vielen Fragen aus verschiedenen Gebieten. Wie 'Trivial Pursuit'. (s/w)

Diskette 122
Spiel
-**PD_BOLO**: Sie kennen Arkanoid, dann sollten Sie auch PD_Bolo kennen. Interessante Break-Out-Variante.

Diskette 123
Utilities
-**HYPERFORMAT**: Formatierprogramm von Claus Brod.
-**FSELECT**: Die neue Fileselectorbox. Wird fest installiert. Kommt bei jedem Aufruf. Wer sich noch mit der alten herumquält ist selbst d'ran schuld.
-**DISKMON**: Diskettenmonitor. (s/w)
-**PC_ASS**: Assembler für Sharp-Taschencomputer.

Diskette 125
Zeichenprogramm
-**CAD 2**: Objektorientiertes Grafikprogramm. z.B. mit Zoom ohne Auflösungsverlust.(s/w)

Diskette 126
-**CUBE HACK**: Terminalprogramm mit integrierter, umfangreicher Programmiersprache.
-**OR SPREAD**: Spreadsheetähnliches Programm zur Linearoptinmierung, Netzplanentwicklung, Kürzeste-Wege-Berechnung. (s/w)

Diskette 128
Spielesammlung
-**HORRORSCHLOß**: Der Weg durch viele Gänge verbirgt viele Gefahren.
-**INVERSI**: Das bekannte Spiel, gegen den Rechner oder zu zweit. (s/w)
-**INVATION**: Space Invadors mit 3-D Vektorgrafik. (s/w)
-**ZARGE**: Ein 32000K-Bild wird gepuzzelt (s/w)

PUBLIC DOMAIN KLASSIKER

Wir haben Ihnen hier einige der besten PD-Disketten aufgeführt, die trotz Alter bislang unübertroffen sind.

Diskette 129
Utilities
-**GFA-SHELL**: Shell für GFA-Basic, incl. CLI und Sprite-Editor (s/w)
-**OMIKRON-SHELL**: Shell für OMIKRON.-Basic System. Aufruf von Interpreter, Compiler, Linker. Generiert PRG, TOS, TTP oder ACC. (s/w)
-**DISKSAVE**: Speichert Bildschirm per 'Alt+Help' auf Disk.
-**APPLE-ST**: Datenübertragung Apple II --> ATARI ST.

Diskette 130
Utilities
-**HFORMAT**: Formatierprogramm als Accessory.
-**RESCUE**: Rettet gelöschte Files. (s/w)
-**DISKINFO**: Gibt Directory sortiert auf Drucker (Liste oder Label) bzw. Datei aus. (s/w)
-**ACCLOADER**: Ermöglich die Auswahl bei mehr als 6 Accessories.(s/w)

Diskette 132
Simulationen
-**WATOR**: Simulation eines biologischen Systems, am Zusammenleben von Haien und Fischen.
-**LIFE**: Ein sehr flexible Life-Version. (s/w)
-**QUICKLIFE**: Rasend schnelles Life. (s/w)

Machen Sie mit

Möchten Sie ein selbstgeschriebenes Programm in die PD Sammlung geben? Kein Problem. Schicken Sie es uns auf einer Diskette zu, samt einer Bestätigung, daß es von Ihnen erstellt wurde und frei von Rechten Dritter ist.

Diskette 133
-**CODEWARS**: REDCODE-Programm. Der Kampf im Rechner. Incl. ausführlich Einführung in die Theorie.
-**KERNE 2.1**: Erweiterte Version der 'Krieg der Kerne'.

DISKETTE 134
Spielesammlung
-**GALERIE**: Wunderschöne Patience-Version. Für kalte Frühlings-Abende.
-**KREUZ AS**: Umsetzung des Spielautomaten. Mut zum Risiko ist angesagt. (s/w)

Diskette 142
-**SCANNER**: Das Softwarepaket zum Scannerprojekt in ST-Computer 4/88. Umrechnung des Bildes (Helligkeit und Kontrast) auch im Nachhinein, div. Bildgrößen, Grauwertberechnung und was sonst noch zum Scannen benötigt wird.

Diskette 143
Utilities
-**MS-DOS STEP**: Schaltet B-Laufwerks-Step auf 6ms. Für 5.25"-Laufwerke im 40-Spur-Betrieb. Als ACC, PRG.
-**BITTE EIN BIT 3.0**: Schnelles Kopierprogramm als Accessory. Hohe Geschwindigkeit durch Ignorieren leerer Tracks. Automatische Formaterkennung.
-**FORMAT 1.0**: Formatierprogramm als Accessory. Formatiert im Hintergrund (Multitasking).
-**SUPER-FILECOPY**: Filekopierprogramm. Ideal für ein Laufwerk.
-**MC87_1_9**: Formatier-, Kopier- und Harddiskbackupprogramm.

Diskette 144
Shells
-**GULAM-SHELL**: Die Shell für Modula-Programmierer (siehe ST-Computer 5/88 - Modula-Test).
-**FLESH**: Universelles Shell-Construction-Kit.
-**TEX-SHELL**: Shell für das Satzprogramm ST-TEX.

Diskette 145
Grafik
-**SHAPES**: Das Programm läßt ein animiertes Pferd über eine Landschaft galoppieren.(s/w)
-**MANIPULATOR**: Programm zum Verfremden von Bildern.(s/w)

Diskette 146
Geographieprogramme
-**GEOGRAPH**: Enthält Daten von 159 Ländern. Hauptstädte, Regierungen, Produkte, Handel, Religion, Energiehaushalt, Landwirtschaft, Währung, u.a.. Abfrage der Daten in Quizform. (s/w)
-**GLOBUS**: Zeigt die Erde von einem beliebigem Punkt aus dem Weltraum.

Diskette 147
Dateiverwaltung
-**DATEI**: Flexibles Datenverwaltungsprogramm. Freie Maskenerstellung per Dialog. Import, Export, Steuerzeichenanpassung.
-**ARTUS**: Artikelverwaltungsprogramm. Dient zum Verwalten von Artikeln aus Zeitschriften. Auswahl nach Stichpunkten, Namen, Monat, u.a..

Diskette 148
Spiele
-**CAVEMINER**: Boulder Dash-Variante. Auf der Suche nach Diamanten. Doch Vorsicht vor Felssteinen, die gerne auf dem eigenen Kopf landen. Überblick, Geschick und Reaktion sind gefragt. (s/w)
-**BUBBLE_EDIT**: Feld-Editor für Bubble Bobble. (s/w)

Diskette 149
Utilities
-**QUICKMOUSE**: Beschleunigt die Mausgeschwindigkeit.
-**KEY HELP**: Ermöglicht die Eingabe eines beliebigen Zeichens (ASCII-Code) nach Drücken von Alternate-Control.
-**DUMP**: Accessory, mit dem eine beliebige Datei angezeigt und einzelne Daten darin geändert werden können.
-**MENÜEDIT**: Eine Art RCS für GFA-Basic-Menüleisten.
Erzeugt einen kompletten Programmkopf für GFA-Basic, zum 'Einmergen' in eigenes Programm. (s/w)
-**TEMPELMON**: Verfeinerte Version des ST-Monitors/Debuggers.

Diskette 150
Spiele
-**DRACHEN**: Umsetzung eines 2000 Jahre alten Brettspiels. Grafisch sehr schön gestaltet. (s/w)
-**SHANG**: Ähnlich Drachen, unterschiedlicher Bildaufbau. (s/w)

Diskette 151
Fractals
-**3D-FRACTALS**: Programm erzeugt ein dreidimensionales Fractalbild. Sehr interessantes Programm. (s/w)

Diskette 152
Etikettendrucker
-**STICKER**: Disketikettendrucker mit sehr vielen Optionen. Liest Directory, übernimmt Programmnamen ins Etikett. Etikett mit Bild. Integriertes Zeichenprogramm für Etikett-Bilder. (s/w)

Diskette 153
Puzzlespiel
-**PENTIMO**: Verschiedene Puzzleteile müssen durch Drehen und Verschieben in einem beliebig großen Feld plaziert werden. Sehr schön gestaltet. (s/w)

Diskette 154
Anwendersoftware
-**SCHECK**: Druckprogramm für Schecks und Überweisungsaufträge. Leicht bedienbar. Anpassung an jedes Formular. Speicherbare Stammdaten (Kontonummer, Bankleitzahl, Namen, etc.).
-**DRUCKER**: Druckeranpassungsprogramm für SCHECK.

Zeichenerklärung:

s/w = nur monochrom
f = nur Farbe

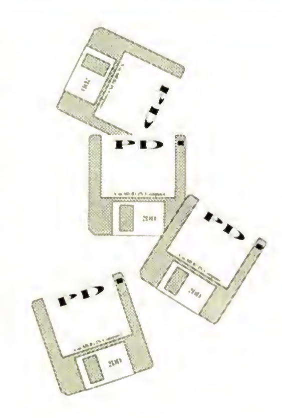
Sonderdisks

Die folgenden Programme sind nicht Public-Domain.
Sie können aber bei uns bezogen werden.

A.) **TOS**: Das Betriebssystem auf Diskette vom 6.2.1986. Was tun, wenn Programme mit Blitter-TOS nicht laufen? Einfach das alte TOS von Disk laden.
Unkostenbeitrag DM 15.-

B.) **RCS**: Das Resource-Construction Set aus dem ATARI-Entwicklungspakets. Unverzichtbar bei der GEM-Programmierung.
Unkostenbeitrag DM 15.-

Bis auf die Höhe des Unkostenbeitrages gelten die gleichen Versandbedingungen wie bei der Public-Domain Software. (siehe vorherige Seite)





STARKE SOFTWARE

SALIX-PROLOG

leistungsfähiges KI-Programmiersystem
- Edinburgh-Standard
- schnell, ca. 1200 LIpS
- ca. 160 eingebaute Funktionen
- Gleitkommaarithmetik
- läuft unter GEM
- ca. 140 GEM Library Funktionen
- Datenbankeditor
- Einbindung von Fremdeditoren
- zyklische Strukturen werden verarbeitet
- Exception Handling
- leistungsfähiges Testsystem

Jetzt neu: Version 2.6
- benutzerdefinierte Funktionen
- globale Variable
- Clipboard-Device

DM 198,–

Update von 2.0 – 2.5 DM 49,–

SALIX DIALOGEXPERTE

NEU

Wer schon einmal eine Anwendung unter GEM geschrieben hat – mit Drop-down-Menüs, Dialogboxen und Alerts, wird es wissen:
So einfach sich die einzelnen Objekte mit dem Resource Construction Set Editor erzeugen lassen, so umständlich ist ihre Einbettung in das Anwendungsprogramm.
Es geht auch anders: SALIX hat jetzt einen DIALOG-Experten entwickelt, der die RSC- und die DEF-Dateien selbständig untersucht. Dann geht's erst richtig los: **„Programming by doing"** heißt die Device. Zunächst werden die Fenster der Anwendung definiert – mit der Muas. Dann wird die Bedeutung der Menüpunkte festgelegt, einfach indem man sie anklickt und das aufzurufende Prädikat eingibt.
Mehrere Menüleisten sind zulässig. Die einzelnen Menüpunkte lassen sich in logische Gruppen zusammenfassen, das Check-Symbol wird kontrolliert, und und und...
Die Knöpfe der Dialogboxen und Alerts lassen sich symbolisch ansprechen, eine Reihe von Prädikaten ermöglicht den einfachen Zugriff auf die GEM-Objekte von der Anwendung aus, komplexe Formulare können auch dynamisch aus der Anwendung heraus aufgebaut werden.
Sie haben die Wahl: sich weiterhin damit abzuquälen, Menüpunkte auszuzählen, oder sich lieber auf das Design und Logik Ihrer Anwendung zu konzentrieren.
Der Dialogexperte läuft unter SALIX PROLOG 2.6.

DM 79,–

ST-C.A.R.
Computer Aided Regulation

Das Programm zur Systemanalyse und Regelungssynthese

Zielgruppe / Aufgabenbereich
Studenten der Ingenieurwissenschaftlichen Fachrichtung
- Veranschaulichung der Auswirkung von Regelkreisglieder/-strecken
- Überprüfung erarbeiteter Lösungen (numerisch und grafisch)
- Einsatz in der Entwicklung eigener Regelsysteme (z. B. Diplomarbeit)

Ingenieure der technischen Fachrichtung
- Einsatz in der Entwicklung benötigter Regelkreise (Zeitersparnis)
- Systemanalyse vorhandener Regelungen (Untersuchungen möglicher Änderungen)
- Unterstützung beim Entwurf durch 'weniger Erfahrene' (Auffrischung des Wissens, eigenes Wissen überprüfen)

Institute, Lehr- und Forschungseinrichtungen des technisch-wissenschaftlichen Bereiches
- Einsatz in der Entwicklung benötigter Regelsysteme
- Analyse vorhandener Regelsysteme (Untersuchung möglicher Änderungen)
- Hilfe bei der Erstellung von Lehr-(Unterrichts-)Material
- Direkter Einsatz im Unterricht ('am lebenden Objekt')

C.A.R. ...das Programm mit umfassender Leistung
C.A.R. ...ausführliche Anleitung mit Beispielen
C.A.R. ...das Programm für professionelle Ansprüche

DM 198,–

ST PRINT

Neue Version

DAS VIELFACH BEWÄHRTE MULTIACCESSORY JETZT IN ERWEITERTER VERSION V2.1 INCL. VIRUSKILLER

ST-PRINT erleichtert durch seine Vielseitigkeit und einfache Bedienung die tägliche Arbeit. Insbesondere Anpassungs- und Verständigungsprobleme zwischen Computer und Drucker werden gelöst.

1. Resetfeste Ramdisk
- Größe einstellbar von 32 – 4000 KB, auf Wunsch resetfest oder abschaltbar
- kann auf Laufwerk C bis P gelegt werden
- arbeitet problemlos mit einer Harddisk zusammen

2. Druckerspooler
- Größe einstellbare von 2 – 510 KB
- abschaltbar
- arbeitet mit TOS- und GEM-Programmen
- auch für Hardcopies
- in Maschinensprache programmiert und interruptgesteuert, dadurch immer optimale Geschwindigkeit beim Ausdruck

3. Druckervoreinstellung
- komfortable Druckereinstellung mit der Maus: Knopfdruck statt Handbuch oder DIP-Schalter
- viele Einstellmöglichkeiten: Zeilenvorschub, Schriftart, Zeichensatz, Papierränder etc.
- Einstellung des Druckers vom Desktop und aus jedem GEM-Programm (VIP-Professional, Wordplus, Tempus etc.) möglich
- **kann an alle Matrixdrucker angepaßt werden, diverse Druckertreiber gehören zum Lieferumfang, auch eigene Anpassungen können erstellt werden**
- **Druckeranpassung jederzeit nachladbar**

4. Druckerzeichenkonverter
- Anpassung aller Atari-Zeichen an den verwendeten Drucker, ermöglicht den korrekten Ausdruck von Umlauten und Sonderzeichen auf allen Matrixdruckern
- Umlaute und Sonderzeichen können gemischt verwendet werden
- endlich problemloser Ausdruck mit Programmen (TOS oder GEM), die keine eigene Druckeranpassung haben

5. Hardcopyroutine
- nutzt die Fähigkeiten von 9-, 18- und 24-Nadeldruckern
- kann an jeden Matrixdrucker angepaßt werden
- verschiedene Auflösungen, Schnelldruck bis Qualitätsdruck
- arbeitet mit dem eingebauten Spooler zusammen *gespoolte Hardcopy
- Auslösung der Hardcopy durch Alternate/Help, kann genauso wieder abgebrochen werden
- Umsetzung der Farben in Grauwerte (im MID-RES Modus)

DM 69,–

Update DM 19,–

ST DIGITAL
LOGIKSIMULATOR FÜR DEN ATARI ST

Ein Programm zum Erstellen, Testen und Analysieren von Logikschaltungen für Ausbildung und Hobby-Elektronik.
- Komfortable GEM-Umgebung
- Bauteile lassen sich per Maus plazieren u. verdrahten
- Umfassendes Bauteile-Set (Grundgatter, Ein-/Ausgabe-Bausteine, Flip-Flops, etc.)
- Definition zusätzlicher Bauteile durch Makrotechnik
- Makros können in Libraries gespeichert werden
- Interaktive Simulation mit Darstellung der Leitungszustände (d. h. Betätigung von 'Schaltern' mit der Maus und sofortige Reaktion der Schaltung)
- Erzeugen von Impulsdiagrammen
- Hardcopy-Funktion
- Hardwarevoraussetzung: ST/Monochrom-Monitor
- gängige Druckertreiber vorhanden

DM 89,–

ST ANALOG

NEU

SIMULATION VON ANALOGSCHALTUNGEN

Das Simulations-Programm zum Analysieren, Testen und Entwickeln von analogen Elektronikschaltungen (Kettenschaltungen) für Hobby, Ausbildung und Studium.
- Komfortable Maus-Steuerung
- Grafischer Schaltungsaufbau
- Einfache Eingabe und Änderung von Bauteilwerten
- Max. 65 Bauteile pro Schaltung (z. B. Stromquellen, Übertrager, Schwingkreise, offene und kurzgeschlossene Stichleitungen, Übertragungsleitungen usw.)
- „Wobbel-Generator" von 1 Hz bis 2 GHz
- Max. 0,1 Hz bzw. 1 Hz Auflösung
- Wahlmöglichkeit zwischen sehr schneller oder sehr genauer Berechnung
- Grafische Ausgabe von Spannungs- und Stromverhältnissen, von Phasenverläufen und von Eingangsimpedanzen
- logarithmische und lineare Koordinatenachsen
- Verstärkung und Offset einstellbar
- Einfaches Testen der Schaltung im Rückwärtsbetrieb
- Digitale Anzeige von Funktionswerten bei diskreten Frequenzen
- Hardcopy-Funktion
- Hardwarevoraussetzung: ST mit Monochrom-Monitor
- ausführliches deutsches Handbuch

DM 98,–

ST-MATH
DAS PROGRAMM FÜR SYMBOLISCHE ALGEBRA UND ANALYSIS

- rechnet ohne Rundungsfehler
- verarbeitet symbolische Ausdrücke wie $2\times + 3y = 5z$
- löst Gleichungen nach beliebigen Variablen auf
- beherrscht Grenzwerte, Differential- und Integralrechnung
- ideal für Matritzenrechnung
- unglaublich schnell, da **vollständig** in Assembler geschrieben
- nicht nur ein Mathematik-Programm, sondern gleichzeitig eine komplette KI-Sprache, die leicht erlernbar ist
- die Kombination von Mathematik-System und Programmiersprache ermöglicht auch Anfängern komplexe Mathe-Programme mühelos zu schreiben
- einfache und komfortable Bedienung
- für Schüler, die sich Fehlrechnungen ersparen wollen
- für Lehrer, die mit ST-MATH Klausuren mühelos und schnell korrigieren wollen
- für Studenten, die lange Umformungen und Rechnungen zeitsparend durchführen wollen
- für Ingenieure und sonst. Anwender, die oft komplexe nichtnumerische Probleme lösen müssen
- für Jeden, der eine preiswerte, aber dennoch vollwertige Sprache für künstliche Intelligenz sucht, die leicht zu erlernen ist
- Hardware: ATARI ST mit 512 KB und ROM-TOS oder 1 MB (dann auch RAM-TOS möglich), eins. Disk.-Laufw.
- arbeitet mit Farb- und SW-Monitoren
- Lieferung mit umfangreichem deutschen Handbuch, das die Mathematikfunktion detailliert erläutert und zugleich eine ausführliche Einführung in die Programmierung von ST-MATH enthält.

Ein Spitzenprogramm DM 98,–

* alle Preise sind unverbindlich empfohlene Verkaufspreise

BESTELL-COUPON an Heim-Verlag, Heidelberger Landstraße 194, 6100 Darmstadt-Eberstadt

Bitte senden Sie mir: ______________________

____

zzgl. DM 5,- Versandkosten (unabhängig von bestellter Stückzahl)
☐ per Nachnahme ☐ Verrechnungsscheck liegt bei

Name, Vorname ______________________

Straße, Hausnr. ______________________

PLZ, Ort ______________________

Benutzen Sie auch die in ST COMPUTER vorhandene Bestellkarte.

Heim Verlag
Heidelberger Landstraße 194
6100 Darmstadt-Eberstadt
Telefon 0 61 51 - 5 60 57

In der nächsten ST-Computer lesen Sie unter anderem

ATARIs CD-ROM im Test

Vor einem Jahr wurde es angekündigt; jetzt ist es da. Das "Optofile CDAR 504" von ATARI kann nicht nur das schweizer Telefonbuch lesen, sondern auch die Dire Straits spielen. Was sich hinter dem grauen Gehäuse versteckt, erfahren Sie in der nächsten Ausgabe.

Zwei auf einmal

In unserm nächsten Hardwareprojekt geht's um den ROM-Port des ST. Als erstes wird der ROM-Port mit einer Treiberkarte gepuffert, so daß weitere Peripheriekarten problemlos angeschlossen werden können. Gleichzeitig stellen wir unsere neue ROM-Karte vor, mit der man bis zu 1,5 MByte an ROM Kapazität gewinnt. Als besonderen leckerbissen verfügt die Karte über einen 36-Bit breiten I/O-Port. Es lohnt sich rein zu schauen.

Face To Face: GFA versus Omikron.

BASIC, die meist verbreitete Programmiersprache bei Personal Computern, ist bei den ST Rechnern stark vertreten. Die zwei Konkurrenten GFA und Omikron. bringen neue leistungstärkere und Versionen ihrer Interpreter. Beide standen bei uns auf dem Prüfstand.

Es ist Show Time

Zum zweiten mal veranstaltet ATARI in Düsseldorf seine eigene Messe. Vom 2. bis 4. September werden auf dem Messegelände in der Stadt am Rhein hunderte verschiedener Produkte für den ATARI ST sowie dessen kleinen Bruder der XL-Serie gezeigt. Wir werden ausführlich berichten.

Änderungen vorbehalten !

Die ST-Computer Ausgabe 10 '88 erscheint wegen der Sommerpause am 30.9.1988

Fragen an die Redaktion

Ein Magazin wie die ST-Computer zu erstellen, kostet sehr viel Zeit und Mühe. Da wir ja weiterhin vorhaben, die Qualität zu steigern (ja, auch wenn das manchmal danebengeht), haben wir Redakteure ein großes Anliegen an Sie, liebe Leserinnen und Leser:

Bitte haben Sie Verständnis dafür, daß Fragen an die Redaktion nur Donnerstags von 1400-1700 Uhr telefonisch beantwortet werden können.

Vielen Dank für Ihr Verständnis

Ihre Redaktion

Impressum ST Computer

Chefredakteur:
Uwe Bärtels (UB)

Redaktion:
Uwe Bärtels (UB)
Harald Egel (HE)
Marcelo Merino (MM)
Harald Schneider (HS)

Redaktionelle Mitarbeiter:
C.Borgmeier (CBO), Claus Brod (CB), Stefan Höhn (SH), Raymund Hofmann (RH), Oliver Joppich (OJO), Jürgen Leonhard (JL), Claus P. Lippert (CPL), Markus Nerding (MN), Chr. Schormann (CS), Andreas Suchy (AS), Jörg Wilhelm (JW), Fernando Brand (FB)

Autoren dieser Ausgabe:
M.Backschat, D.Brockhaus, I.Brümmer, A.Esser, S.Geier, W.Keller, H.U.Mayer, Th.Moltzen, K.E.Neugebauer, D.Rabich, Chr.Schmitz-Moormann (CSM), H.Scholpp, H.D.Schultz (HDS), K.Steger, R.Tolksdorf, E.Winkler, C.D.Ziegler

Redaktion: "Merlin" Computer GmbH
Postfach 59 69
Industriestr. 26
6236 Eschborn
Tel.: 0 61 96/48 18 11
FAX : 0 61 96/4 11 37

Verlag:
Heim Fachverlag
Heidelberger Landstr. 194
6100 Darmstadt 13
Tel.: 0 61 51/5 60 57
FAX : 0 61 51/5 56 89 + 5 60 59

Verlagsleitung:
H.J.Heim

Anzeigenverkaufsleitung:
U.Heim

Anzeigenverkauf:
K.Magaritis

Anzeigenpreise:
nach Preisliste Nr.3, gültig ab 1.1.88
ISSN 0932-0385

Grafische Gestaltung:
Bernd Weber
Karstin Feist
Bela Computer GmbH

Fotografie:
Bela Computer GmbH, R.Spirandelli, Archiv

Produktion:
K.H.Hoffmann, B.Failer, S.Failer

Druck:
Ferling Druck

Lektorat:
V.Pfeiffer

Bezugsmöglichkeiten:
ATARI-Fachhandel, Zeitschriftenhandel, Kauf- und Warenhäuser oder direkt beim Verlag

ST Computer erscheint 11 x im Jahr
Einzelpreis: DM 7,-, ÖS 56,-, SFr 7,-
Jahresabonnement: DM 70,-
Europ. Ausland : DM 90,-
Luftpost : DM 120,-

Manuskripteinsendungen:
Programmlistings, Bauanleitungen und Manuskripte werden von der Redaktion gerne angenommen. Sie müssen frei von Rechten Dritter sein. Mit seiner Einsendung gibt der Verfasser die Zustimmung zum Abdruck und der Vervielfältigung auf Datenträgern dem Heim Verlag.
Honorare nach Vereinbarung. Für unverlangt eingesandte Manuskripte wird keine Haftung übernommen.

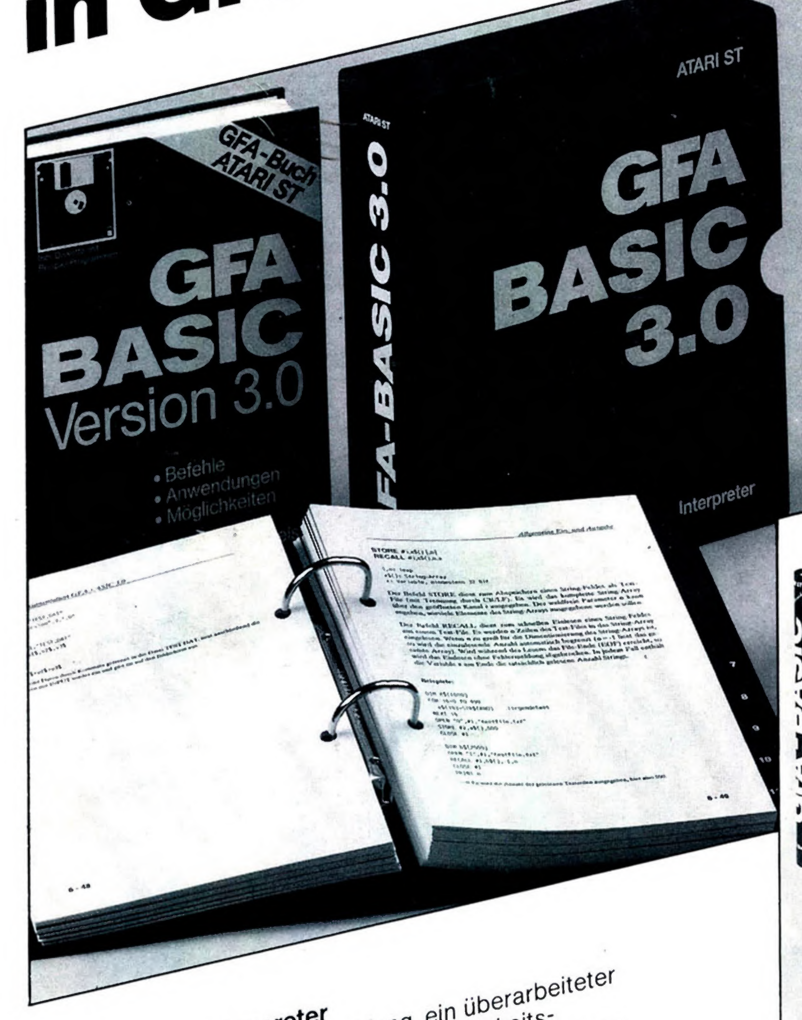
Programmieren in GFA-BASIC

GFA-BASIC 2.0 Interpreter
Das meistverkaufte Programm für ATARI-ST zum Einsteigerpreis DM 99.-

GFA-BASIC 2.0 Compiler
Mit diesem Compiler werden Ihre GFA-BASIC 2.0-Programme noch schneller und sind selbständig lauffähig. DM 99.-

GFA-BASIC-Buch
Frank Ostrowski über sein GFA-BASIC. Sie erhalten viele wertvolle Tips vom Autor des GFA-BASIC. Besonders geeignet für den fortgeschrittenen Anwender.
ISBN 3-89317-001-4 DM 79.-
288 Seiten, inkl. Diskette

GFA-BASIC-Programmierung
Von der Idee über den Entwurf bis hin zum fertigen Programm. Sie lernen mit Hilfe dieses Buches das strukturierte Programmieren in GFA-BASIC.
ISBN 3-89317-003-0 DM 49.-
288 Seiten, inkl. Diskette





GFA-BASIC 3.0 Interpreter
Ein stark erweiterter Befehlsumfang, ein überarbeiteter Editor, eine erneute, deutliche Geschwindigkeitssteigerung und ein komplett neu erstelltes, 550-seitiges Handbuch: Das sind die herausragenden Merkmale der neuen Version 3.0 von GFA-BASIC. DM 198.-

GFA-BASIC-Buch: Version 3.0
Für alle, die bisher mit GFA-BASIC gearbeitet haben und nun den optimalen Umstieg auf die neue Version 3.0 suchen, haben die Handbuch-Autoren Gottfried P. Engels und Markus C. Görgens dieses Buch geschrieben. Anhand vieler Beispiele werden die Möglichkeiten der neuen Anweisungen von GFA-BASIC 3.0 gut verständlich vermittelt. DM 59.-
ISBN 3-89317-004-9
400 Seiten, inkl. Diskette

GFA-CLUB
GFA-ST/PC-Software
bitte Info anfordern
...Anruf genügt.

GFA Systemtechnik GmbH
Heerdter Sandberg 30
D-4000 Düsseldorf 11
Telefon 0211/588011

